

# Swarm

## Level 1b Processor Algorithms

National Space Institute  
Technical University of Denmark

---

**DOC. NO. & ISSUE:** SW-RS-DSC-SY-0002, Issue 6.14



**DATE:** 2021-02-09

**PREPARED BY:** Jonas Bregnhøj Nielsen

**APPROVED BY:** Lars Tøffner-Clausen

**AUTHORIZED BY:** Poul Erik Holmdahl Olsen

**CLASSIFICATION:** Restricted: For use within the  
ESA Swarm project only

This page is intentionally left blank

## DOCUMENT CHANGE LOG

Issue	Change References	Issue Date	Pages Affected	Remarks	Init.
1		2006-09-15	All	Initial Issue	LTC
2		2006-11-01	9 – 35	Added Section 4 on global processing overview. Updated figures and descriptions accordingly. Various minor updates	LTC
3		2007-02-05	All	Added missing sections Major update of Sections 4 and 5 especially related to instrument transfer functions Added appendices	HF LTC
4		2007-04-27	All	Updated according to Level 1b Requirements Review – in particular: Added Section 5: Level 0 Processing Replaced inclusions of documents in Appendices E through J by links	
5		2007-09-07	All	Various minor changes to complete Level 1b Requirements Review.	LTC FCH
5.1		2007-10-05	38-41,43, 46,49-51, 56-57, 62-63,69, 79-83,	Updated calculation of magnetic stray fields, updated EFI CCDB parameters, added fundamental constants to CCDB, and a few minor corrections	LTC FCH
5.2		2008-03-10	Many	Updated calculation of flags. Adopted changes to CCDB – in particular look-up tables and filters. Added new Appendix E with associated (and other) functions. Various other updates aimed at Operational Level 1b Procurement KO	LTC

Issue	Change References	Issue Date	Pages Affected	Remarks	Init.
6		2008-11-18	Many	<p>Updated according to implementation plan by L1b S/W supplier.</p> <p>Updated according to changes in CCDB and Level 1b Products.</p> <p>TCF.VFM estimation data selection procedure switched to alternate method.</p> <p>ACC processing revised</p> <p>Added CurrentInfo, StateInfo, and InducedInfo and more Level 0 to Level 1a conversion description</p> <p>Updated according to revised magnetically induced field computation</p>	LTC
6.1		2008-12-12	17,21-28, 39,46, 54-55,66, 69,71-74, 77,83-84, 87,88,125	<p>Added more Level 0 to Level 1a conversions</p> <p>Corrected and updated <math>B_{ambient}</math> for induced stray fields, and various inconsistencies in Current/State/Induced Info description</p> <p>Added missing “.Structure” to CCDB elements</p>	LTC
6.2		2009-07-10	20, 21, 29, 41, 60, 63, 66-68, 73-75, 78-79, 83, 85-87, 94, 95, 121-123, 127-128	<p>Updated STR processing – direction of transformation in data and aberrational correction</p> <p>Update VFM decoding and added VFM DPU identification</p> <p>Updated sample selection for TCF.VFM estimation</p> <p>Increased Mag-C sampling rate to 1 Hz and updated contents</p> <p>Updated EFI section</p> <p>Removed torque parts of Solar radiation pressure from ACC</p> <p>Added leap-second processing description</p>	LTC

Issue	Change References	Issue Date	Pages Affected	Remarks	Init.
6.3		2010-06-29	23-32, most pages 35-79, 109-117, 128-130, 137, 139-140	<p>Adopted to latest structure and changes of CCDB (introduction of L1BP section, re-arrangement of parameters, update of units) including corresponding updates of in particular processing of: battery currents, AOCS torque response, and stray fields.</p> <p>Moved Level 0 to Level 1a conversions from IMDD to CCDB.TMC.</p> <p>Corrected VFM and ASM Data Conversion (Sections 5.4.7 and 5.4.8)</p> <p>Added CCDB.AOCS. Thruster_Settling_Time</p> <p>Added ASM_Freq_Dev computation</p> <p>Huber weights limited to range ]0..1]</p> <p>VFM processing updated: range, temperatures, delays, and flags</p> <p>Corrected STR CHU combination.</p> <p>Updated STR flags.</p> <p>Updated ACC processing, added temperature weights</p> <p>New Tank Level1a data and temperatures computation</p> <p>Corrected description of B-spline knots and acceleration damping weights</p> <p>Corrected various typing errors and some unclear points.</p>	LTC
6.4		2010-12-03	40, 45f, 56-58, 62, 64-71, 126, 138, 152	<p>Clarified selection and application of TCF.VFM records, a.o. changed TCF.VFM<sup>input</sup> into TCF.VFM<sup>init</sup>.</p> <p>Simplified VFM DPU_id handling by rejection of DPU_id-minority data.</p> <p>Modified ASM and VFM</p>	LTC

Issue	Change References	Issue Date	Pages Affected	Remarks	Init.
				<p>outlier detection.</p> <p>Corrected a-priori term, added Huber weight factor (for L1bOP CDR-3), and modified convergence limit in TCF. VFM estimation</p> <p>Changed from linear to cubic interpolation of <math>F_{ASM}</math> for Mag-H.Flags<sub>B</sub> computation (Table 6-8)</p> <p>Corrected small error in gap filling of Filter function</p> <p>Updated reference frame definitions (App C)</p> <p>Updated ACC document reference (App L)</p>	
6.5		2011-06-24	39, 40, 84, 110, 116-118,	<p>Clarification of <math>B_{ambient}</math></p> <p>Harmonized units of Bus currents (changed mA to A)</p> <p>Added information on CCDB. Structure.CrossSection</p> <p>Corrected ACC timestamp (Eq. 8.3-10), centrifugal acceleration (angular velocity), and <math>a_{GG}</math> computations</p>	LTC
6.6		2012-05-08	22 30 36 41, 50, 51 49 68 71 78 77	<p>Specified which products contain leap second data</p> <p>Corrected timestamping of VFM data.</p> <p>Update of HK_Source for VFM temperatures</p> <p>Corrected location of ASM.Filter from CCDB.L1BP to CCDB</p> <p>Specified timestamps (t) for ASMxAUX_1B product</p> <p>Corrected storing of W_scale values as <math>\log_{10}</math>.</p> <p>Specified values for <math>dB_{AOCS}</math> and <math>dB_{other}</math> in Mag-L Product</p> <p>Added Mag-L.Flags<sub>B</sub> value 255</p> <p>Specify STR pair to use if only one pair is inconsistent.</p>	LTC

Issue	Change References	Issue Date	Pages Affected	Remarks	Init.
			79 89 128 130 global	Deleted last paragraph on obsolete STR flag computation Flags <sub>q</sub> clarification and addition of values 35 and 54 Specified spherical coordinates Specified removal of knots for very short data segments Corrected XXXx_AUX1B names to XXXxAUX_1B	
6.7		2014-04-04	59, 59 93, 98 96-97 159-161	Corrected VFM DPU_id value 1 to 2 (Section 6.3.2) SPR-245: Clarification of GPSR_r and ARP_r CCDB parameters for GPSR (Section 8.1) Clarification of calculation of pitch, roll, and yaw for GPSR processing (Section 8.1.3) Updated references to TII, LP, and GPSR algorithm documents (Appendix I, J, and K)	LTC
6.8	<a href="#">AI-261</a>  <a href="#">SPR-39</a>  <a href="#">AI-259</a>  <a href="#">AI-225</a>	2014-08-29	38 43 57 63 120	Adjusted VFM sampling frequency to <i>approximately</i> 50 Hz (Table 6-1) Corrected condition for $I_{AOCs,VFM}$ , Eq. (6.1-27b) Corrected weights for ASM B-Spline interpolation, Eq. (6.2-190) Specified rejection of initial VFM samples, Section 6.3.3 Corrected sign of acceleration corrections, $a_{GG}$ and $a_{centr}$ , Eqs. (8.3-50) and (8.3-60).	LTC

Issue	Change References	Issue Date	Pages Affected	Remarks	Init.
6.9	<a href="#">AI-286</a>	2014-12-09	30	Corrected EFI timestamps	LTC
	UPID-23 <sup>1</sup>		36-37	Added STR false leap second correction	
	UPID-17		55	Clarified computation/ interpolation of $dF_{AOCS}$ and $dF_{other}$ for Mag-C and Mag-L products	
	UPID-18		76	Corrected which VFM samples form the basis for Mag-H	
	UPID-19		73	Added sanity-check on residual scaling in VFM sampling reduction	
	UPID-20		79	Clarified time window for computing Mag-L.FlagSPlatform	
	UPID-21		79, 87	Added lower limits on attitude uncertainties	
	UPID-24		138	Modified formula for <i>fade-out</i> B-spline knots to account for short data sequences	
		65	Corrected typo (extra “(“)		
6.10	UPID-25	2015-07-13	68-69	Added $d\mathbf{B}_{Sun}$ stray field	LTC
	UPID-26		75	Generation of Mag-C product when no ASM data	
	UPID-2		85-86	Include BBO and lost-in-space STR samples	
	UPID-31		138	B-spline knot generation for very short segments	
		159, 160	Updated versions of EFI algorithm documentation		
6.10a		2015-10-20	62	Added $d\mathbf{B}_{Sun}$ stray field to table of outputs	JBN
			76	Added $d\mathbf{B}_{Sun}$ , $\mathbf{B}_{pre}$ , alpha and beta to table of MAG-C product elements	
6.10b		2015-12-07	69, 72-73	Added $s_{\beta}$ and $s_{exp}$ to Eqs. (6.3-180), (6.3-230) and (6.3-280)	JBN

<sup>1</sup> UPID refers to planned update ID's in Section 3 of SW-PL-DTU-GS-008, "Planned Updates for Level 1b"



Issue	Change References	Issue Date	Pages Affected	Remarks	Init.
6.10c		2017-01-27	49 76 79 140-141	Corrected $F_{err}^{(3)}$ equation in Figure 6-4 Corrected Mag-C $F_{error}$ description in Table 6-7 Added footnote about 1% slack in Table 6-10 at Mag-H/L $Flags_{Platform}$ values 4 and 128 Section 8.4.3.4: Corrected “min” to “max” in text. Added “if statement” and corrected “with” and “W” calculation in text and MATLAB code.	JBN
6.10d		2018-05-17	21 30, 34, 36 69, 72-73 80, 84 56-58, 86-89, 138-140 121 21, 22, 91-94 91-94 99-103	Added SC_XDYN_1B to Table 4-2 Added Level 0 data; HK_xNOM_0_ (PID=4) SID=46 and SID=47 to Table 5-1 and Sections 5.4.6 and 5.4.10 Added $s_t$ , $s_{t2}$ , $ds_{CSC}$ , $ds_{CDC}$ and $ds_{EU}$ to Eqs. (6.3-180), (6.3-230) and (6.3-280) Added Input Parameters for Temperature Correction to Section 6.4.1 together with a new Section (6.4.2.3) on <i>Temperature Correction</i> B-spline interpolations of the ASM (Section 6.2.4.5) and STR (Section 6.4.2.5) data together with the B-spline knot generation (Appendix D 8.4.3.2) revised Replaced Thru_Acc_On with dt_thr, f_thr and thr_set in Output Table in Section 8.3.2 TII removed from Plasma Data Processing EFI Section (Section 7) revised GPSR Algorithm (Section 8.1.3) revised	JBN

Issue	Change References	Issue Date	Pages Affected	Remarks	Init.
			118-128	Level 1b.ACC split into Level 1b.ACC and Level 1b.SC DYN. Accelerometer – ACC Section (Section 8.3) revised	
6.10		2018-07-06	58, 88  91  123  1	Corrected missing levelling terms in Eq's (6.2-240) and (6.4-260)  EFI Section (Section 7) introduction revised  Clarified details in ACC processing (Figure 8-7 and preceding text)  Signed	LTC     PEHO
6.11	e-mail JBN > EQ on 14 February 2019 3:49 PM	2019-02-14	129-132	Section 8.4 ICRF/ITRF Coordinate Transformation restored	JBN
6.12		2020-04-16	85  89	Eq. 6.4-99c and Eq. 6.4-100 revised and Eq. 6.4-105 added in Section 6.4.2.4  Descriptions of Flag_q values 16, 17 and 18 revised to be consistent with AD02 (Table 6-12)	JBN
6.13		2020-06-17	14, 21, 29, 36	Added ASM Burst mode processing notes: [AD05a], Figure 4-2, and Section 5.4.8	LTC
6.14		2021-02-09	54-56  54, 68  21	Added $\Delta \mathbf{B}_{\text{Sun,ASM}}$ correction of the ASM measurements: Eq's (6.2-95), (6.2-97), (6.2-100), (6.2-145), (6.2-152) and (6.2-153)  Added optional filtering of both $\Delta \mathbf{B}_{\text{Sun,ASM}}$ and $\Delta \mathbf{B}_{\text{Sun,VFM}}$ : Eq's (6.2-97) and (6.3-134a)  Added the L2 POD product as a alternative input to $\mathbf{r}$ in Figure 4-2	JBN

<b>Table of Contents</b>	<b>Page</b>
1 Introduction .....	13
1.1 Scope .....	13
2 Applicable and Reference Documents .....	14
2.1 Applicable Documents .....	14
2.2 Reference Documents .....	15
3 Introduction .....	17
3.1 Terminology.....	17
4 Processing Overview .....	19
4.1 Detailed Data Flow .....	21
4.2 Re-processing.....	23
4.3 Time-series .....	24
4.4 Corrections.....	24
4.5 Timestamps .....	25
4.6 Leap Seconds .....	25
5 Level 0 Processing .....	27
5.1 Quality and Validity Checks .....	27
5.2 Data Coordination and Combination .....	28
5.3 Datation of Data .....	29
5.4 Conversions from Level 0 to Level 1a .....	31
5.5 Level 1a Data Description.....	37
6 Magnetic and Attitude Data Processing.....	38
6.1 Magnetic Stray Fields .....	40
6.2 Absolute Scalar Magnetometer – ASM.....	45
6.3 Vector Field Magnetometer – VFM.....	59
6.4 Star Tracker – STR.....	80
7 Electric Field Instrument - EFI.....	91
7.1 Introduction.....	91
7.2 Input and Output Parameters.....	91
7.3 Algorithms .....	94
8 Position and Acceleration .....	95
8.1 GPS Receiver - GPSR .....	95
8.2 Orbit Determination.....	103
8.3 Accelerometer – ACC.....	118
8.4 ICRF/ITRF Coordinate Transformation .....	129

## Appendices

<b>A.</b>	Abbreviations & Acronyms .....	133
<b>B.</b>	Quaternions .....	134
<b>C.</b>	Reference Frames.....	136
<b>D.</b>	B-Splines .....	138
<b>E.</b>	Miscellaneous Functions .....	147
<b>F.</b>	ASM Algorithm Descriptions .....	156
<b>G.</b>	VFM Algorithm Description .....	157
<b>H.</b>	STR Algorithm Description .....	158
<b>I.</b>	EFI-TII Algorithm Description .....	159
<b>J.</b>	EFI-LP Algorithm Description.....	160
<b>K.</b>	GPSR Algorithm Description .....	161
<b>L.</b>	ACC Algorithm Description .....	162

# 1 Introduction

## 1.1 Scope

The present document is prepared as part of the Swarm Level 1b Data Processing specification. The description of the algorithms for the Level 1b processing is contained herein.

## 2 Applicable and Reference Documents

### 2.1 Applicable Documents

The documents listed below are applicable in their latest issue and revision.

- AD01 Swarm Level 0 Products  
Doc. No. SW.IF.EAD.GS.00017  
EADS Astrium
- AD02 Swarm Level 1b Product Definition  
Doc. No. SW-RS-DSC-SY-0007  
National Space Institute, Technical University of Denmark
- AD03 Swarm Level 1b Processor Characterisation and Calibration Data Base  
Doc. No. SW-TN-DSC-SY-0005  
National Space Institute, Technical University of Denmark
- AD04 Swarm Level 1b Processor Justification  
Doc. No. SW-TN-DSC-SY-0003  
National Space Institute, Technical University of Denmark
- AD05 Swarm ASM level 1B algorithms  
Doc. No. AUT/ALGO1B/001  
LETI, France
- AD05a 1 Hz Scalar Data from ASM Burst Mode – Algorithm and Prototype Processor  
Doc. No. SW-CEA-AUT-ALGO1B-008  
LETI, France
- AD06 Vector Field Magnetometer for the Swarm Project, Users Manual  
Doc. No. SW-MA-DTU-VF-5001  
Danish Technical University
- AD07 micro Advanced Stellar Compass User's Manual  
Doc. No. ASC-DTU-MA-3001  
Danish Technical University
- AD08 Swarm CEFI Level 1b Processor Definition  
Doc. No. SW-TN-UOC-EF-003  
University of Calgary, Canada
- AD09 Swarm CEFI-LP Level 1b Algorithms  
Doc. No. SW-TN-IRF-EF-003  
Swedish Institute of Space Physics
- AD10 Swarm Mathematical Conventions  
Doc. No. SW.TN.EAD.SY.00031  
EADS Astrium, Germany
- AD11 Swarm GPSR TE-12 Instrument L1b Algorithms Definition  
Doc. No. SW-TN-SES-GP-0018  
Saab Space AB, Sweden
- AD12 Swarm Phase B, Level 1b Processor, MOD Processing  
Doc. No. SW-TN-DSC-SY-0015  
National Space Institute, Technical University of Denmark

- AD13 Accelerometer Instrument Level 1b Algorithms Definition  
Doc. No. SW-DF-VZL-AC-0001  
VZLU, Czech Republic
- AD14 Swarm Level 1A Product Definition  
Doc. No. SW-ID-GMV-GS-0003  
GMV Aerospace and Defense, Spain

## 2.2 Reference Documents

- RD1 Calibration of the Ørsted Vector Magnetometer  
Nils Olsen et.al.  
Earth Planets Space, 55, 11–18, 2003  
[http://www.spacecenter.dk/~nio/papers/EPS\\_55010011.pdf](http://www.spacecenter.dk/~nio/papers/EPS_55010011.pdf)
- RD2 Data Sampling Rate Reduction for the OERSTED Geomagnetic Satellite  
P.B. Stark  
<http://www.stat.berkeley.edu/~stark/Preprints/Oersted/writeup.htm>
- RD3 Optimal combination of quaternions from multiple star cameras  
L.J. Romans  
[ftp://podaac.jpl.nasa.gov/pub/grace/doc/quaternion\\_memo.pdf](ftp://podaac.jpl.nasa.gov/pub/grace/doc/quaternion_memo.pdf)  
**Note:** The definition of quaternions used in this description differs from the one used here. Our definition has the real part as fourth component, Romans uses a definition with the real part as first component. <http://www.iers.org/>
- RD4 International Earth Rotation and Reference System Service Conventions 2003  
by D.D. McCarthy and G. Petit  
<http://www.iers.org/>
- RD5 Quaternion Toolbox  
Jay St. Pierre  
<http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=1176>
- RD6 Robust Estimation of a Location Parameter  
P. J. Huber  
Annals of Mathematical Statistics, Vol. 35, No. 1, pp. 73-101, March 1964  
<http://links.jstor.org/sici?sici=0003-4851%28196403%2935%3A1%3C73%3AREOALP%3E2.0.CO%3B2-K>
- RD7 A Practical Guide to Splines  
Carl de Boor  
Applied Mathematical Sciences, vol. 27, Springer 1978  
ISBN 0-387-95366-3
- RD8 Observing, Modeling, and Interpreting Magnetic Fields of the Solid Earth  
Mioara Manda and Michael Purucker  
Surveys in Geophysics, vol. 26, pp. 415–459, Springer 2005
- RD9 IERS Conventions web page  
<ftp://maia.usno.navy.mil/conv2000/chapter5> (Last visit 27. March, 2007)

- RD10 IAU Standards of Fundamental Astronomy software collection  
[http://www.iau-sofa.rl.ac.uk/2005\\_0901.html](http://www.iau-sofa.rl.ac.uk/2005_0901.html) (Last visit 27. March, 2007)
- RD11 RINEX: The Receiver Independent Exchange Format Version 3.00  
by Werner Gurtner, Astronomical Institute, University of Bern  
Dated: 2006-12-19  
<http://igscb.jpl.nasa.gov/igscb/data/format/rinex300.pdf>
- RD12 Earth Explorer Mission CFI Software  
EXPLORER\_LIB SOFTWARE USER MANUAL  
Doc. No. CS-MA-DMS-GS-0003  
DEIMOS Space S.L., Spain



### 3 Introduction

The Swarm Level 1b Processor processes the Swarm Level 0 data [AD01] from each of the three satellites individually and produces the Swarm Level 1b Data Products [AD02]. This document contains the detailed description of the algorithms required for the processing.

The processing of Swarm Level 1b data falls into three groups: magnetic data, plasma data, and ephemeris (positional) data. The content of this document is as follows:

This section introduces some terminology.

Section 4 contains an overview of the overall processing and describes the concepts of spatial/temporal synchronisation and time-series.

Section 5 contains the description of the processing of Level 0 data into Level 1a data together with a description of the Level 1a data used in the subsequent sections.

Section 6 contains the description of the processing of magnetic and attitude data.

Section 7 contains the description of the plasma data processing.

Section 8 contains the description of position, velocity and acceleration (GPSR and ACC) data processing.

#### 3.1 Terminology

The general terms and typography used in this document is listed in Table 3-1. See also the “Swarm Mathematical Conventions” document [AD10].

<b>v</b>	Boldfaced terms indicates vectors of any dimension (typically 3)
$\mathbf{v}_{\text{frame}}$	A subscript of a vector is used to denote the reference frame of the vector
$ \mathbf{v} $	Length of a vector, $ \mathbf{v}  = \mathbf{v} \cdot \mathbf{v}$
$\mathbf{e}_v$	Unit vector along $\mathbf{v}$ , $\mathbf{e}_v = \mathbf{v}/ \mathbf{v} $
$\overline{\overline{M}}$ or $\overline{M}$	Double lines above terms indicates matrices – or they are not marked specifically, but it should be clear from the context, that it is a matrix
<b>a</b>	Acceleration
<b>B</b>	Magnetic field vector
<b>E</b>	Electrical field vector
<b>F</b>	Magnetic field intensity, aka scalar field
<b>I</b>	Current (electrical), often a vector ( <b>I</b> )
<b>q</b>	Quaternion, a 4 dimensional vector defining a rotation/transformation. See Appendix B and [AD10]

<b>r</b>	Position
<b>t</b>	Time
<b>t<sub>0,XXX</sub></b>	Timestamp of XXX measurement in Level 0 telemetry (see Section 4.5)
<b>t<sub>XXX</sub></b>	Time of XXX measurement, $t_{XXX} = t_{0,XXX}$ corrected for delays except group delays of filtering of measurements
<b>t<sub>out,XXX</sub></b>	Time of XXX measurement adjusted for all delays, $t_{out,XXX} = t_{XXX} - \text{filter group delay}$
<b>T</b>	Temperature
<b>v</b>	Velocity
<b>ρ</b>	Density
<b>×</b>	Matrix multiplication operator, or quaternion transformation operator. If $\mathbf{q}_{B \leftarrow A}$ is a quaternion and $\mathbf{v}$ a 3 dimensional vector, $\mathbf{v}_B = \mathbf{q}_{B \leftarrow A} \times \mathbf{v}_A$ . $\mathbf{q} \times \mathbf{v} \equiv \mathbf{q}^* \otimes \mathbf{v} \otimes \mathbf{q}$ – see also Eq. (B-20) in Appendix B.

Table 3-1 Terms

## 4 Processing Overview

The overall processing of Swarm Level 1b data is outlined in Figure 4-1. The telemetry inputs in the top of the figure indicate the instrument or subsystem supplying the basic data streams for the processing. Secondary inputs such as house-keeping telemetry and auxiliary information are not shown in the figure. The outputs in the bottom of Figure 4-1 are the Level 1b Data Products (see Table 4-2) generated by the processing blocks. Some of the Level 1b data elements are fed back into other processing boxes shown by the green arrows.

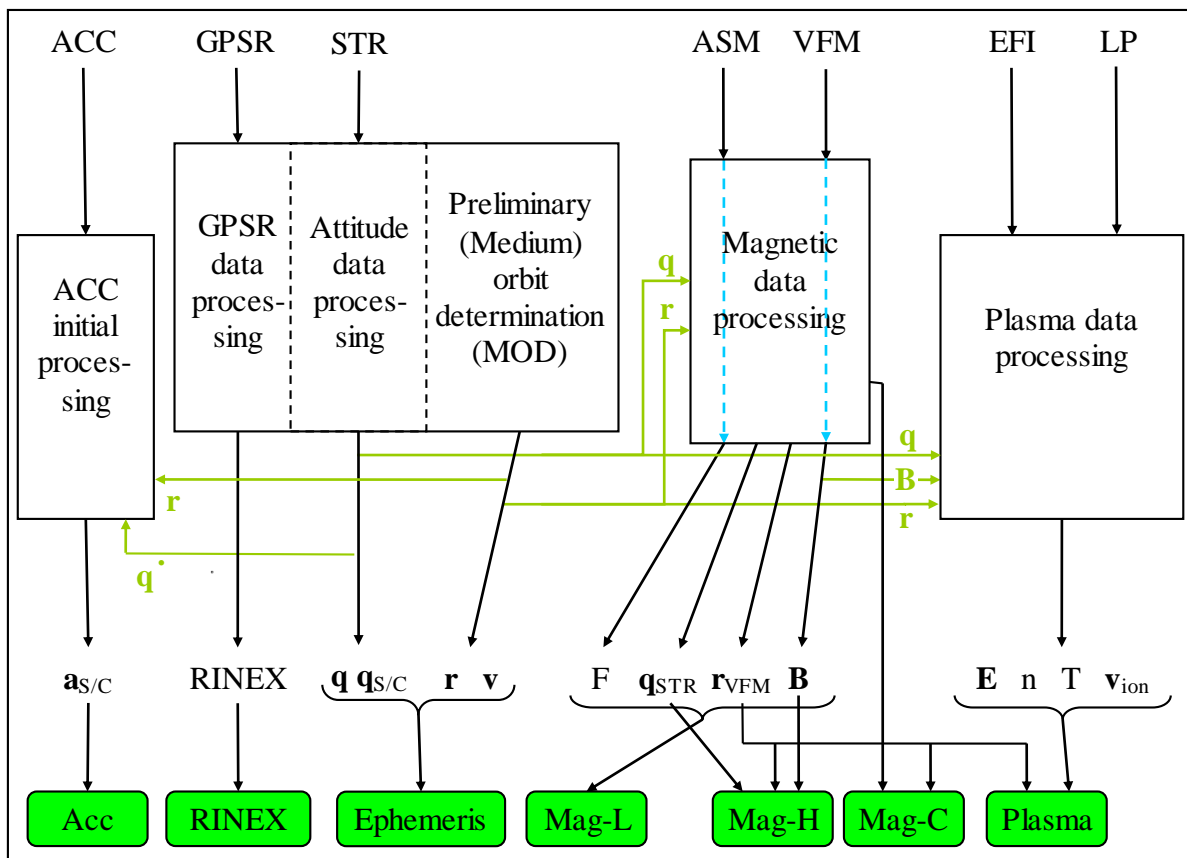


Figure 4-1 Overall Level 1b Processing

The processing sequence is as follows:

1. Process GPSR and STR data, and perform Medium accuracy Orbit Determination (MOD), cf. Sections 8.1, 6.4 and 8.2.)
2. Process magnetic (ASM and VFM) data processing, cf. Section 6.
3. Plasma data processing (Section 7)
4. ACC data processing (Section 8.3)

Note: The attitude information generated in Step 1. is being interpolated in steps 2.-4. The algorithm for this interpolation is described in Section 6.4.2.

The Level 1b Data Products are 1 day (0:00 – 24:00 UTC) in length, which fits well with the MOD and magnetic data processing, hence data are most conveniently processed in batches of one day. However, to avoid edge effects in various filter and interpolation processes the processing interval is expanded some minutes in both ends.

The basic input data streams and their data rates and cut-off (-3 dB) frequencies are listed in Table 4-1.

Instrument	Rate	Cut-off	Comment
GPSR	0.1 Hz		
ACC	1 Hz	TBD	To be provided
ASM	1 Hz	ca. 0.4 Hz	Butterworth filter, 2 <sup>nd</sup> order, 3 in cascade
VFM	50 Hz	ca. 9 Hz	Bessel filter, 4 <sup>th</sup> order, $f_0 \approx 22$ Hz
STR	1 Hz	0.44 Hz	Box-car filter, i.e. the STR provides attitudes averaged over the image integration time (1 second), 6 dB/octave
LP	2 and 16 Hz	TBD	To be provided
TII	2 and 16 Hz	TBD	To be provided

Table 4-1 Input Data

Table 4-2 below provides some brief information on the Level 1b data products. To facilitate the use of the magnetic data from all three Swarm satellites the Mag-L (as well as Ephemeris) data are synchronized to integral UTC seconds.

Product	ShortName	Rate	Comment
GPSX_RO_1B GPSX_RN_1B	RINEX	0.1 Hz	GPSR data converted to RINEX format
MODX_SC_1B	Ephemeris	1 Hz	Position, velocity, and attitude from GPSR, STR and MOD
ACCX_PR_1B	Acc	1 Hz	Pre-calibrated non-gravitational acceleration
SC_XDYN_1B	Dyn	1 Hz	Auxiliary data for precise orbit determination and acceleration modelling
MAGX_LR_1B	Mag-L	1 Hz (UTC)	Magnetic vector and scalar data in VFM and NEC frames (of $\mathbf{r}_{VFM}$ ) including $\mathbf{q}_{NEC \leftarrow CRF}$ transformation quaternions and $\mathbf{r}_{VFM}$ position.
MAGX_HR_1B	Mag-H	50 Hz	Magnetic vector data in VFM and NEC frames at intrinsic VFM sampling time instants.
MAGX_CA_1B	Mag-C	0.25 Hz	Magnetic <i>calibration</i> data to monitor, verify and re-estimate CCDB.VFM.Cal and TCF.VFM parameters.
ASMXAUX_1B	Asm_Aux	50 Hz	Magnetic stray fields at ASM sensor
VFMXAUX_1B	Vfm_Aux	50 Hz	Magnetic stray fields at VFM sensor

Product	ShortName	Rate	Comment
EFIX_LP_1B	Plasma	2 Hz	Provided by the EFI package, the LP instrument

Table 4-2 Level 1b Data Products, cf [AD02]

### 4.1 Detailed Data Flow

The detailed data flows from Level 0 and auxiliary products to the Level 1b products are shown in the following figures. The “ISP Decoding” is described in Section 5.

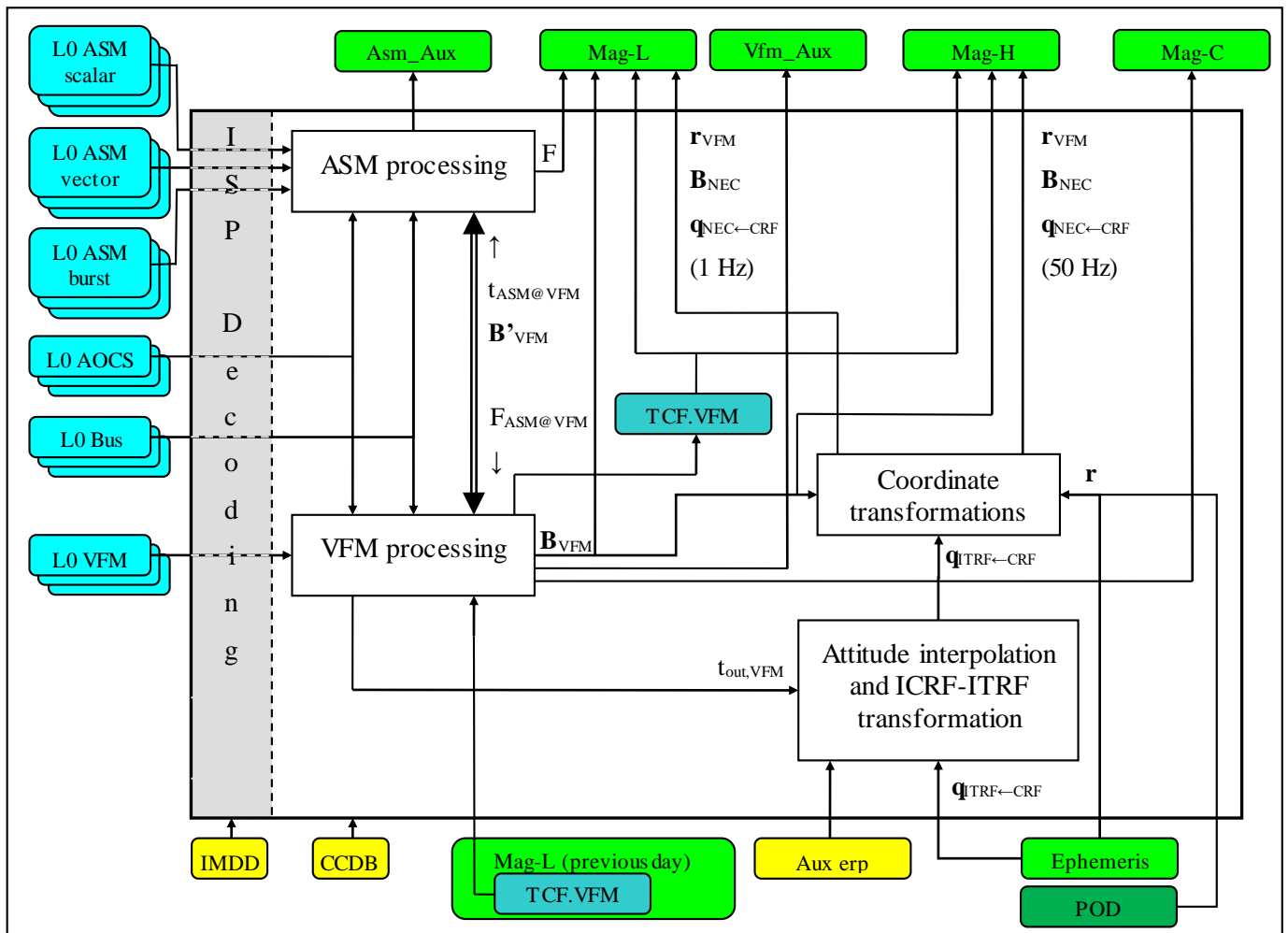


Figure 4-2 Magnetic Data Processing

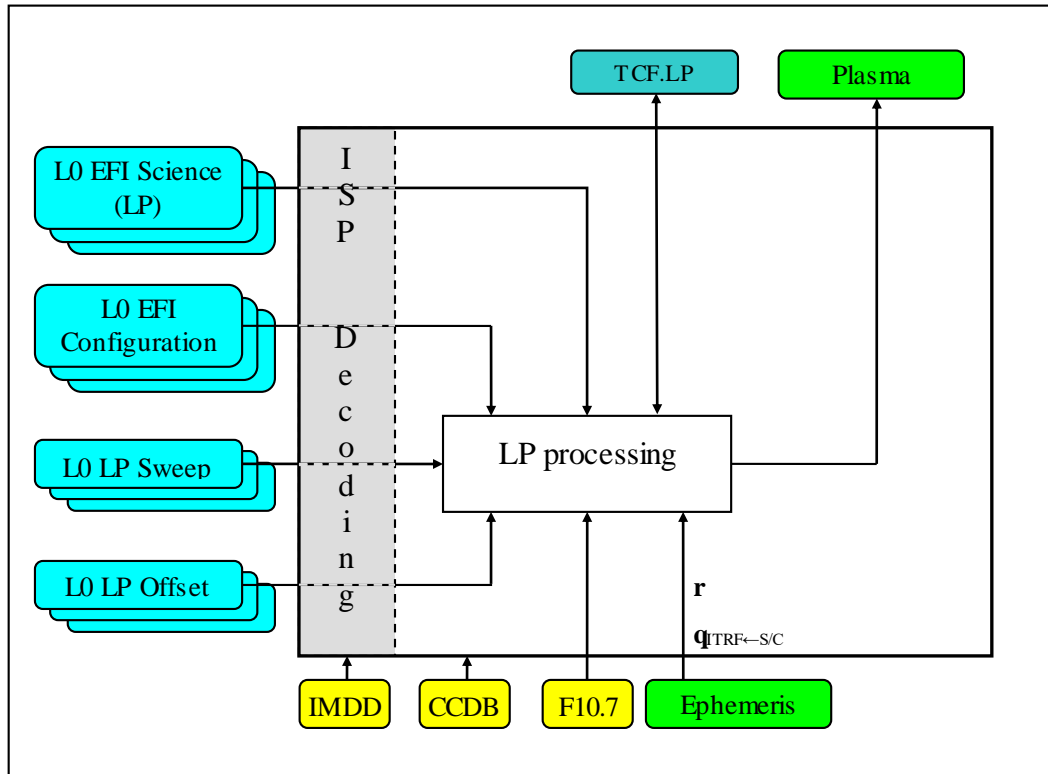


Figure 4-3 Plasma Data Processing

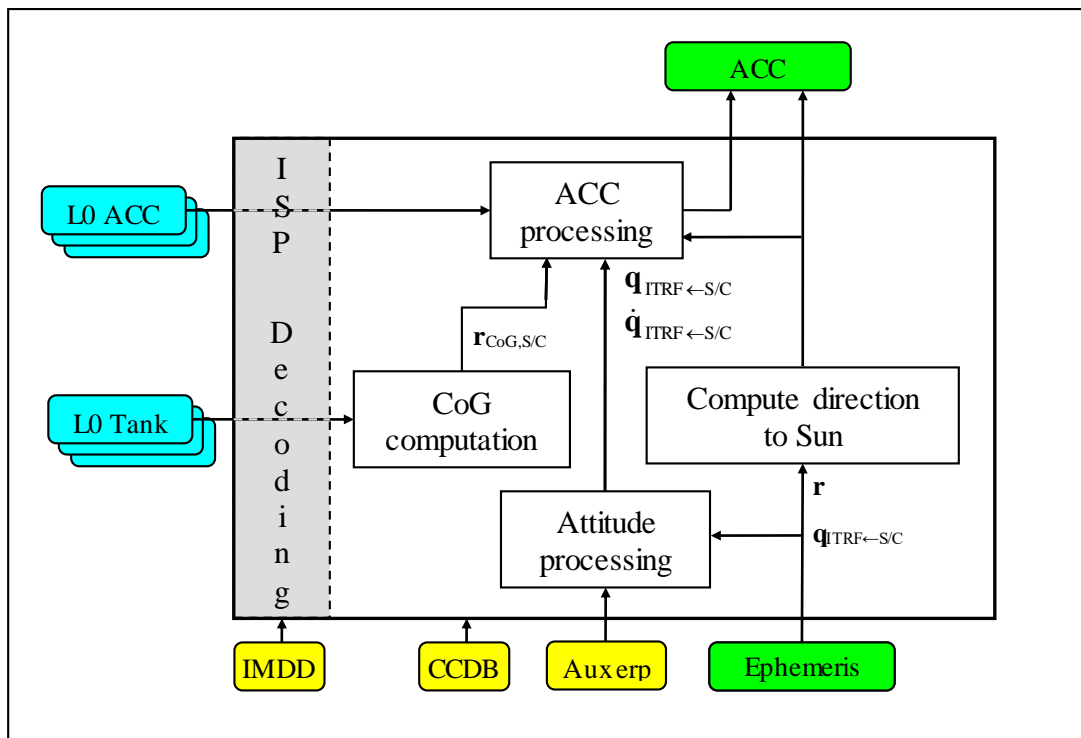


Figure 4-4 Accelerometer Data Processing

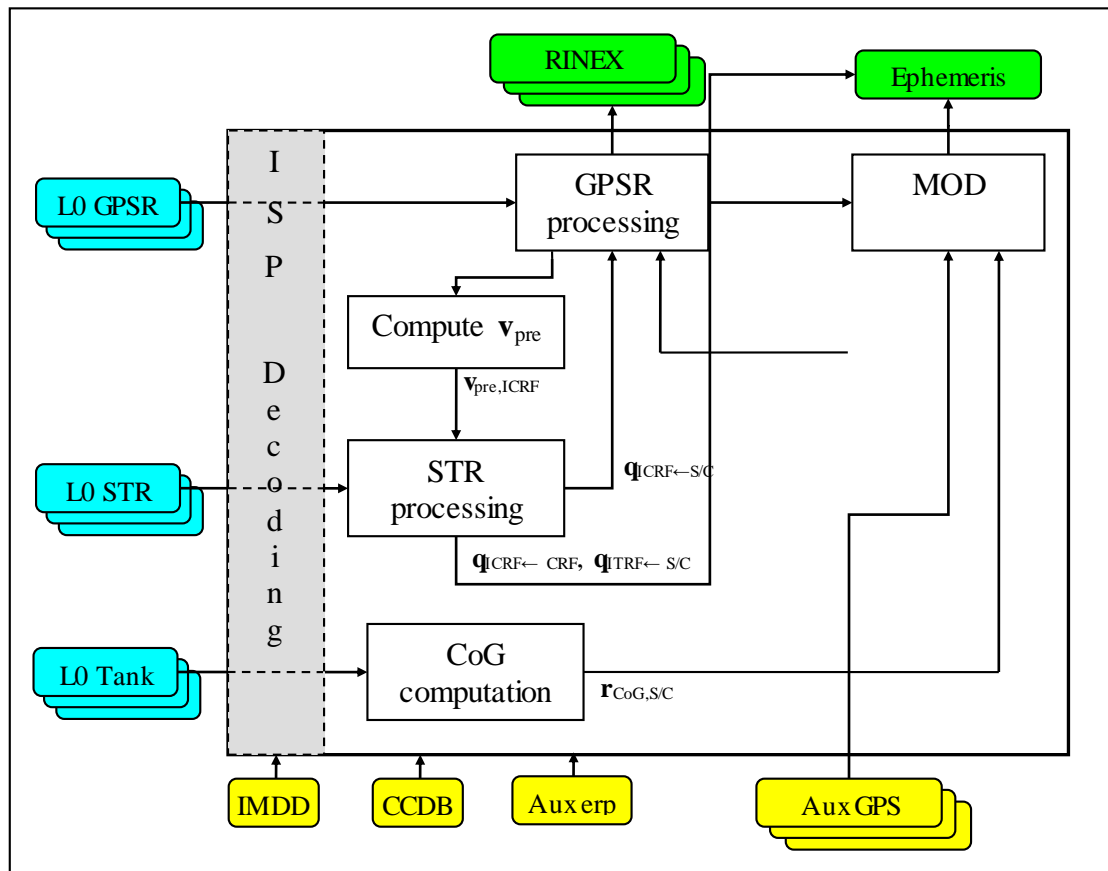


Figure 4-5 GPSR Data and MOD Processing

## 4.2 Re-processing

No automatic re-processing of data is foreseen. The processing steps shown in Figure 4-1 are performed only when all required telemetry and auxiliary data are available. Manual re-processing is possible for all chains and may typically be effected as a result of an update of CCDB parameters.

A dedicated re-processing chain for the magnetic data is foreseen. The CCDB.Structure.**STR\_q\_VFM** alignment quaternion parameter is expected to be updated 2-5 times throughout the Mission. This calls for a recomputation of the  $\mathbf{B}_{NEC}$  elements of the Mag-L and Mag-H Level 1b Products, which may be accomplished either by a full reprocessing (Figure 4-2) or by a simple recomputation of  $\mathbf{B}_{NEC}$  data as depicted in Figure 4-6 below (and described in Section 6.3.4, Eq. (6.3-420) on Page 77). This re-processing step generates new versions of the Mag-L and Mag-H Level 1b Products.

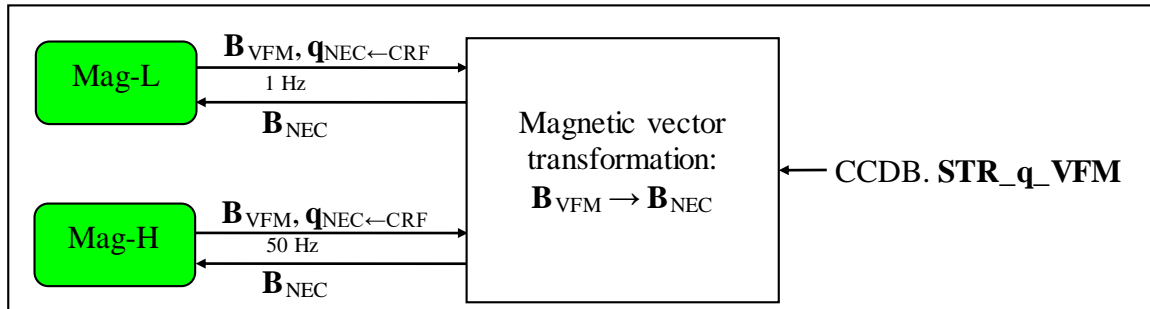


Figure 4-6 Reprocessing after CCDB.Structure.STR\_q\_VFM update

### 4.3 Time-series

The Swarm data consist of time-series, i.e. sequences of typically evenly time-spaced observations (measurements) of continuous, physical signals. The data rates and the exact time-instants of the measurements provided by the instruments do typically not match those of the final Level 1b products. This is handled by various advanced interpolation methods described in the detailed processing descriptions in later Sections.

The primary source of a given time-series is one instrument, e.g. the VFM supplies the Level 1b.Mag-H. $\mathbf{B}$  magnetic vector data – see also the dashed lines in Figure 4-1. However, the processing of the data involves the combination of the primary source with various other (secondary) time-series: e.g. several house-keeping (HK) channels such as temperatures and currents, S/C attitude, position and velocity, and possibly data from other instruments. The secondary data are typically provided at different rates and different time-instants than the primary observations, hence they need to be processed: typically interpolated and possibly filtered to the time-instants of the primary time-series. Furthermore, most of the measurements are discrete samples of continuous signals, hence the real physical signal can only be approximated. Data with a moderate requirement on accuracy and typically slow changes are linearly interpolated as this is a simple and stable solution. Scientific measurements with high accuracy demands are interpolated using higher order polynomials (position) or advanced, parameterized smoothing B-splines to achieve the most accurate re-sampled values (scalar magnetic data and attitude). Data from the magneto-torquers of the AOCS (Attitude and Orbit Control System) sub-system are processed specifically with respect to the primary data source (VFM or ASM) as the requirement on the knowledge of the AOCS signals is high and – contrary to other HK data – the AOCS signals are controlled and monitored carefully.

### 4.4 Corrections

The magnetic and acceleration measurements are affected by “S/C disturbances” which need to be accounted for to get the true, ambient values of these quantities. This is done by computing the actual disturbances and subtracting these from the measurements, i.e.  $y_{\text{clean}} = y_{\text{measured}} - y_{\text{disturbance}}$ , for any quantity  $y$ . Disturbance is sometimes also referred to as “correction”, i.e.  $y_{\text{clean}} = y_{\text{measured}} - y_{\text{correction}}$ , and  $y_{\text{correction}} = y_{\text{disturbance}}$ .



## 4.5 Timestamps

Telemetry data is – one way or the other – timestamped on-board the satellite. This timestamp is transmitted in the telemetry and termed *Level 0 timestamp*,  $t_{0,XXX}$  (this is identical to the Level 1a timestamp). Generally, this timestamp is somehow affected by a delay compared to the true time of the measurement; the true time is referred to as  $t_{XXX}$ .  $t_{XXX} = t_{0,XXX} - \text{delay}$ . See Figure 4-7. Observe that this delay may be positive (the timestamp is after the actual measurement time) or negative (measurement is after timestamp). Furthermore, the measurement itself may be delayed with respect to the actual physical signal being measured. E.g. a thermistor measuring the temperature of some component may experience a delay during temperature changes while heat flows between the component and the thermistor; or a filter within the instrument may introduce a delay of the output data from the instrument – see Figure 4-7. The time of the original signal corresponding to a measurement is termed  $t_{out,XXX}$ ;  $t_{out,XXX} = t_{XXX} - \text{filter delay}$ .  $t_{out,XXX}$  is the timestamp used in the Level 1b data products as this most accurately<sup>2</sup> reflects the time of the signal given by the measurement.

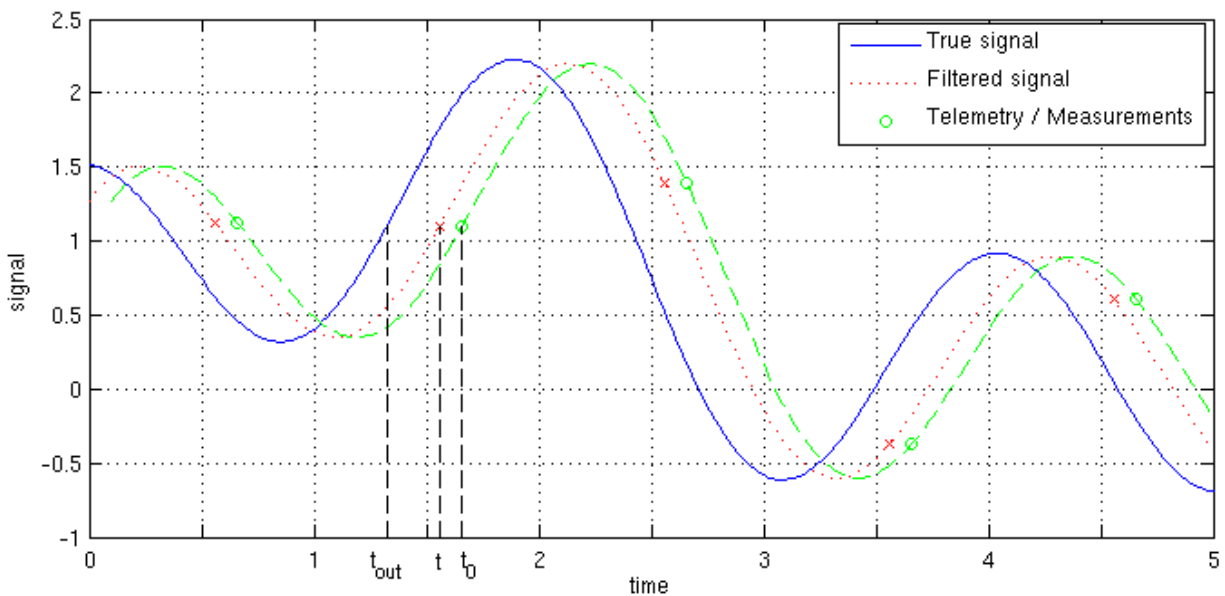


Figure 4-7 Example of timestamps

## 4.6 Leap Seconds

Leap Seconds are introduced to keep the time frames UTC and UT1 aligned within 1 second. They are introduced by International Earth Rotation and Reference Systems Service (IERS) when appropriate either by the end of 30 June or end of 31 December. Positive leap seconds are introduced as an extra second denoted “23:59:60”, negative leap seconds are introduced by omitting the last second of the day, i.e. “23:59:59” does not exist in this case.

When leap seconds are introduced, GPSR and MOD processing, interpolation of ASM and STR data (Sections 6.2 and 6.4) and resampling of VFM data (Section 6.3) need to account

<sup>2</sup> The filter delay may depend on the signal frequency; however this is ignored in the Level 1b products.

for the leap second. For all other data streams, the introduction of leap seconds may be ignored. GPSR and MOD leap second processing is described in the respective documents ([AD11] and [AD12]). For ASM, VFM, and STR leap second introduction is handled by assuming the length of this particular day to be 86,401 seconds (or 86,399 seconds in case a negative leap second is introduced). E.g. during the resampling or interpolation, data are time-stamped using second-of-the-day, and data from the day following the leap-second start at 86,401 (or 86,399). Similarly when processing data of the day immediately after the leap-second, data from the day of the leap-second shall be shifted 86,401 (86,399) seconds.

For products using GPS timestamps (GPSx\_RO\_1B, GPSx\_RN\_1N, and MODx\_SC\_1B) leap second data are preserved. For all other products, i.e. products using UTC timestamps, leap second data are omitted.

## 5 Level 0 Processing

The Swarm measurement data are received as Level 0 Product Files [AD01]. These files contain data sets with sequentially ordered and duplicate free Instrument Source Packets (ISP) from individual Application Processes, e.g. ASM scalar mode data. However, there may be gaps within the Level 0 files. These gaps are attempted to be covered by renewed download of the data resulting in new Level 0 files, which may contain ISPs included in previous Level 0 files. Figure 5-1 illustrates a possible scenario of four Level 0 files from a given application process. The blue bars show the periods of data (time) covered by four files; the shaded parts indicate gaps in the files. The net effect of these dumps is a small gap in the data just prior to the data in the 4<sup>th</sup> (bottom) file.

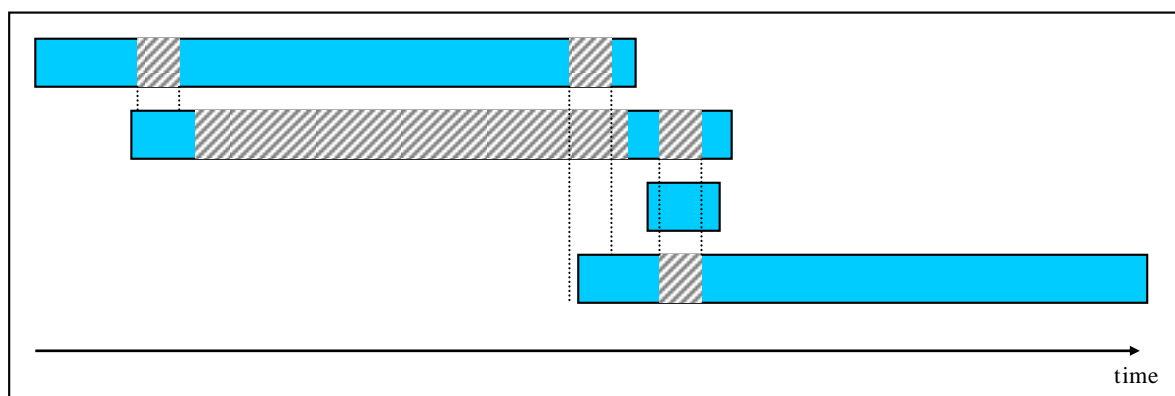


Figure 5-1 Scenario of Level 0 Products with gaps

When the Level 1b Processor is requested to process a given period of data, appropriate and available Level 0 Products containing data from the given period are provided to the Processor, hence more than one product for each Application Process may be supplied. Hence the Processor must merge the Level 0 files into a single stream of sequential, duplicate- and (as far as possible) error-free list of ISPs without gaps for each Application Process. This includes a decoding of the ISP header to identify the proper sequential order, and may further involve additional checks of error control and/or validity fields. See Sections 5.1 and 5.2.

The ISPs are decoded into Level 1a data (Section 5.4.7) using the Instrument Measurement Data Definition (IMDD) for format definition/decoding and the CCDB.TMC elements for conversion from the raw Level 0 telemetry into the units of the Level 1a data. The definitions of the ISPs can be found in [AD01]. The Level 1a data may be stored in Level 1a Product Files according to the configuration of the Level 1b Processor.

### 5.1 Quality and Validity Checks

The ISPs shall be checked for quality and validity. This includes checks of

- Error control code. If the check of the error control code fails, the packet shall be marked as invalid and shall be rejected.
- PPS synchronization source.
  - If the source is OBC a warning flag is to be set in the Level 1a and Level 1b Products. The processing shall be done as for GPS PPS.

- If the source is not GPS or OBC the packet shall be marked as invalid and shall be rejected.
- PPS synchronization quality
  - If only one of PPS and spacecraft O/B Time is received a warning flag is to be set in the Level 1a and Level 1b Products. The processing shall be done as if both are received.
  - If none is received the packet shall be marked as invalid and shall be rejected.
- Validity flags. If the instrument measurements are marked as invalid, the packet shall be marked as invalid and shall be rejected, if not otherwise stated in the detailed algorithms given in the later sections of this document.
- ISP fields for specific characteristics, e.g. validity boundaries. For details refer to the algorithms definitions given in the later sections of this document and the definition of the flags in the Level 1b Product Definition, ref. [AD02].

In case of gaps in the ISP stream (invalid or missing packets) there will be jumps in the Level 1a data, which are handled in the later algorithms.

## 5.2 Data Coordination and Combination

In addition to the simple extraction and conversion of ISP fields some further processing of Level 0 to Level 1a data is required. This includes

- Interleaving and grouping of data
- Handling of different modes
- Collection and adaptation of auxiliary data (including housekeeping and characterisation data)

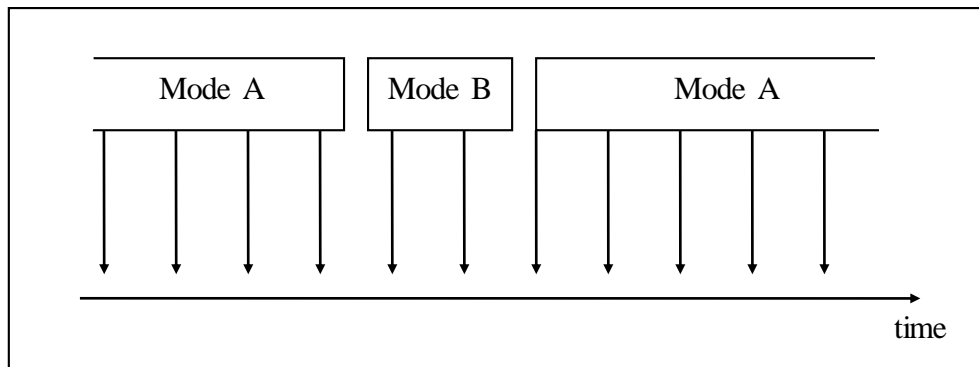
### 5.2.1 Interleaving and grouping of data

Packets with different Process IDs (PID) will be stored in different product types and packets with different structure may be stored in different data sets. The data from these packets need to be interleaved and grouped suitable for further processing.

### 5.2.2 Handling of modes

Some instruments can operate in more than one mode generating measurement data.

Some of the data are only for diagnostic purposes and can be ignored in the Level 1b processing.



Some are providing similar data. The Level 1b processing of these data is identical, where not explicitly defined as to be different. The data of these modes shall be merged. An identification shall be added to Level 1a and Level 1b products identifying the source (mode) of the measurement data.

**ACC:** Only Measurement Mode. Ignore Selftest Mode ISPs

**ASM:** Merge Scalar and Vector Mode products. Ignore SpectroMag, SpectroLaser and SpectroPowerLaser Mode products.

**EFI:** TBD

**VFM:** Merge Compressed and Uncompressed Mode ISPs.

### 5.2.3 Collection and adaptation of auxiliary data

For the processing of the measurement data housekeeping, characterization/calibration and other auxiliary data are needed. These have to be collected from the input files (Level 0, Level 1b and auxiliary products) as needed for the processing.

For some of the data alternative sources are available. These data shall be selected from the nominal source. Only if the nominal source is not available the data shall be read from the alternative source. Example: TCF parameter: Nominal source is Level 1b product. Alternative source is CCDB.

Several units on the satellite are redundant. The characterization and calibration parameters depend on the active configuration. Therefore the characterization and calibration parameter suitable for the active configuration have to be selected.

The housekeeping, characterization/calibration and other auxiliary data have to be selected suitably for the time instant or time interval of the measurements. This may include some minor adaptation, e.g. interpolation of data to the needed time instant.

## 5.3 Datation of Data

The datation of data from the various instruments and Bus house-keeping telemetry is done according to the following table:

System	PID/SID	L0 element	L1a section	Description	t <sub>0</sub> (= Level1a.t)
ACC	61/11	dt	ACC	Delta time since PPS (unit microsecond)	t_pps + dt
ASM	31/11 32/12	dt	ASM	Delta time since PPS (unit millisecond)	t_pps + dt
EFI/LP	56,57,58/-	dt	LP*	Delta time since PPS (unit 1/4096 seconds)	t_pps + dt
EFI/TII	56,57,58/-		TII		t_pps
GPSR	Specific handling of time, described in [AD11]				
STR	80,90/ 3,4,5	t_coi	STR	Time stamp of the centre of integration period, CDS format	t_coi
VFM	41/11 41/14	VFM <sub>time</sub> VFM <sub>PPStime</sub>	VFM	VFM Time of 1 <sup>st</sup> measurement respectively last PPS	t_pps + VFM <sub>PPStime</sub> / 8 · 10 <sup>6</sup>
HK	4/6	t_mtq	AOCS. Mag_HK	MTQ currents 1/2/3 Command Issued (UTC)	t_mtq
	4/6	t_thruster	AOCS. Thru_HK	Thruster activate, Time Command Issued (UTC)	t_thruster
	4/7	-	Bus		t_pps
	4/8	-	AOCS. Prop_HK		t_pps
	4/46		STR.HK		t_pps
	4/47		Bus.OpticalBench		t_pps

Table 5-1 Datation: Level 0 to Level 1a Conversion

## 5.4 Conversions from Level 0 to Level 1a

This Section describes the various conversions of the Level 0 telemetry channels (*raw values*) into the Level 1a values which may be engineering units (*eu* – typically science measurements) or physical units (Volts, temperature, etc. – typically house-keeping data).

### 5.4.1.1 Processing of Bus State and Current Information

The extraction of the relevant state information and current measurements in the Level 0 HK data into Level 1a data suitable for the calculation of magnetic stray fields – steady as well as electrically and magnetically induced fields – is performed using the StateInfo, CurrentInfo, and InducedInfo stored in the CCDB.Bus.State, CCDB.Bus.Current, and CCDB.Bus.Induced arrays respectively. The resulting Level 1a data are listed in Section 5.5.

### 5.4.2 State Information – Static Stray Fields

The Level 0 state information from HK\_xNOM\_0\_ (SID = 7) is processed into the state vector Levell1a.Bus.State.States using the CCDB.Bus.State information. The purpose of this state vector is to allow state dependent remanent stray magnetic fields to be easily computed. The CCDB.Bus.State characterizes the states which change the remanent magnetic field of some instrument or subsystem, and it includes the corresponding changes in the remanent field.

CCDB.Bus.State.States is an array of type StateInfo:

Name	Type	Description												
ID	string	Name of Level 0 Bus channel. Note: the same ID may occur several times and this may also be the ID of a heater or thruster although these states are also given in the <b>Heater_Stat</b> and <b>Thru_Stat</b> vectors of Levell1a.Bus.State.												
type	string	Specifies translation type – how to compute Levell1a.Bus.State.States <sub>i</sub> : <table border="1" data-bbox="459 1370 1321 1998"> <thead> <tr> <th>type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>'Level0'</td> <td>No translation, use value in Level0 directly: States<sub>i</sub> = Level 0.(ID)</td> </tr> <tr> <td>'Equal'</td> <td>States<sub>i</sub> = <math>\begin{cases} 1 &amp; \text{if Level 0.(ID) = value} \\ 0 &amp; \text{otherwise} \end{cases}</math></td> </tr> <tr> <td>'Between'</td> <td>States<sub>i</sub> = <math>\begin{cases} 1 &amp; \text{if value(1) &lt;= Level 0.(ID) &lt;= value(2)} \\ 0 &amp; \text{otherwise} \end{cases}</math></td> </tr> <tr> <td>'Greater'</td> <td>States<sub>i</sub> = <math>\begin{cases} 1 &amp; \text{if Level 0.(ID) &gt; value} \\ 0 &amp; \text{otherwise} \end{cases}</math></td> </tr> <tr> <td>'Less'</td> <td>States<sub>i</sub> = <math>\begin{cases} 1 &amp; \text{if Level 0.(ID) &lt; value} \\ 0 &amp; \text{otherwise} \end{cases}</math></td> </tr> </tbody> </table>	type	Description	'Level0'	No translation, use value in Level0 directly: States <sub>i</sub> = Level 0.(ID)	'Equal'	States <sub>i</sub> = $\begin{cases} 1 & \text{if Level 0.(ID) = value} \\ 0 & \text{otherwise} \end{cases}$	'Between'	States <sub>i</sub> = $\begin{cases} 1 & \text{if value(1) <= Level 0.(ID) <= value(2)} \\ 0 & \text{otherwise} \end{cases}$	'Greater'	States <sub>i</sub> = $\begin{cases} 1 & \text{if Level 0.(ID) > value} \\ 0 & \text{otherwise} \end{cases}$	'Less'	States <sub>i</sub> = $\begin{cases} 1 & \text{if Level 0.(ID) < value} \\ 0 & \text{otherwise} \end{cases}$
type	Description													
'Level0'	No translation, use value in Level0 directly: States <sub>i</sub> = Level 0.(ID)													
'Equal'	States <sub>i</sub> = $\begin{cases} 1 & \text{if Level 0.(ID) = value} \\ 0 & \text{otherwise} \end{cases}$													
'Between'	States <sub>i</sub> = $\begin{cases} 1 & \text{if value(1) <= Level 0.(ID) <= value(2)} \\ 0 & \text{otherwise} \end{cases}$													
'Greater'	States <sub>i</sub> = $\begin{cases} 1 & \text{if Level 0.(ID) > value} \\ 0 & \text{otherwise} \end{cases}$													
'Less'	States <sub>i</sub> = $\begin{cases} 1 & \text{if Level 0.(ID) < value} \\ 0 & \text{otherwise} \end{cases}$													
value	vector	Value(s) for comparison, if type ≠ 'Level0'												

Name	Type	Description
<b>dB_ASM</b>	vector	Change in magnetic stray field vector at ASM sensor position when States <sub>i</sub> changes from 0 to 1
<b>dB_VFM</b>	vector	Similar to <b>dB_ASM</b> for the VFM sensor position

Table 5-2 StateInfo – Elements of CCDB.Bus.State Array

### 5.4.3 Bus Current Measurements

The calculation of the electrically induced magnetic fields uses the information in Levella.Bus.Curr. This is constructed from Level 0 Bus current measurements provided in HK\_xNOM\_0\_ (SID = 7) using the CCDB.Bus.Current (and CCDB.TMC) specification, which lists the appropriate current channels and their stray magnetic field vectors. These stray field vectors are given for unit current (1 A), i.e. they are to be multiplied by the Levella.Bus.Curr value to yield the actual stray magnetic field.

CCDB.Bus.Current consists of three arrays: **I<sub>SP</sub>**, **I<sub>Bat</sub>**, and **I<sub>other</sub>** for the Solar Arrays, Batteries, and others respectively. The array elements are of type CurrentInfo:

Name	Type	Description
ID	string	Name of Level 0 Bus channel.
I_xfer	LutId	Linear look-up table for translating Level 0 to Level 1a value: $\text{Levella.Bus.Curr.I}_{\text{Source}} = \text{Interpol}_{\text{Lin}}(\text{I\_xfer}, \text{Level 0.(ID)})$ The $\text{Interpol}_{\text{Lin}}$ function is given in Appendix E.1
<b>dB_ASM</b>	vector	Electrically induced magnetic stray field vector at ASM sensor position for unit current (1 A), i.e. multiply by Level 1a value
<b>dB_VFM</b>	vector	Same as <b>dB_ASM</b> for the VFM sensor position
<b>dB_Body</b>	vector	Same as <b>dB_ASM</b> for the S/C body.

Table 5-3 CurrentInfo – Elements of CCDB.Bus.Current Array

### 5.4.4 State Dependent Magnetically Induced Fields

Instruments and subsystems with soft-magnetic characteristics (i.e. producing magnetically induced stray fields) are characterized by the CCDB.Bus.Static.Ind<sub>xxx</sub>, CCDB.Bus.State.Induced, and CCDB.AOCS.M<sub>xxx</sub> elements – for the AOCS, the M<sub>xxx</sub> matrices includes both direct and induced effects. The CCDB.Bus.Static and CCDB.AOCS elements contain the properties for static soft-magnetic properties as well as *default* state properties for equipment with non-static characteristics. The characteristics of state dependent soft-magnetic properties are stored in the CCDB.Bus.State.Induced array of InducedInfo (Table 5-4), which is used to generate Levella.Bus.State.IndStates.



Name	Type	Description												
ID	string	Name of Level 0 Bus channel. Note: the same ID may occur several times												
type	string	Specifies translation type – how to compute $Level0.Bus.State.IndStates_i$ : <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>'Level0'</td> <td>No translation, use value in Level0 directly: <math>IndStates_i = Level0.(ID)</math></td> </tr> <tr> <td>'Equal'</td> <td><math>IndStates_i = \begin{cases} 1 &amp; \text{if } Level0.(ID) = \text{value} \\ 0 &amp; \text{otherwise} \end{cases}</math></td> </tr> <tr> <td>'Between'</td> <td><math>IndStates_i = \begin{cases} 1 &amp; \text{if } \text{value}(1) \leq Level0.(ID) \leq \text{value}(2) \\ 0 &amp; \text{otherwise} \end{cases}</math></td> </tr> <tr> <td>'Greater'</td> <td><math>IndStates_i = \begin{cases} 1 &amp; \text{if } Level0.(ID) &gt; \text{value} \\ 0 &amp; \text{otherwise} \end{cases}</math></td> </tr> <tr> <td>'Less'</td> <td><math>IndStates_i = \begin{cases} 1 &amp; \text{if } Level0.(ID) &lt; \text{value} \\ 0 &amp; \text{otherwise} \end{cases}</math></td> </tr> </tbody> </table>	type	Description	'Level0'	No translation, use value in Level0 directly: $IndStates_i = Level0.(ID)$	'Equal'	$IndStates_i = \begin{cases} 1 & \text{if } Level0.(ID) = \text{value} \\ 0 & \text{otherwise} \end{cases}$	'Between'	$IndStates_i = \begin{cases} 1 & \text{if } \text{value}(1) \leq Level0.(ID) \leq \text{value}(2) \\ 0 & \text{otherwise} \end{cases}$	'Greater'	$IndStates_i = \begin{cases} 1 & \text{if } Level0.(ID) > \text{value} \\ 0 & \text{otherwise} \end{cases}$	'Less'	$IndStates_i = \begin{cases} 1 & \text{if } Level0.(ID) < \text{value} \\ 0 & \text{otherwise} \end{cases}$
type	Description													
'Level0'	No translation, use value in Level0 directly: $IndStates_i = Level0.(ID)$													
'Equal'	$IndStates_i = \begin{cases} 1 & \text{if } Level0.(ID) = \text{value} \\ 0 & \text{otherwise} \end{cases}$													
'Between'	$IndStates_i = \begin{cases} 1 & \text{if } \text{value}(1) \leq Level0.(ID) \leq \text{value}(2) \\ 0 & \text{otherwise} \end{cases}$													
'Greater'	$IndStates_i = \begin{cases} 1 & \text{if } Level0.(ID) > \text{value} \\ 0 & \text{otherwise} \end{cases}$													
'Less'	$IndStates_i = \begin{cases} 1 & \text{if } Level0.(ID) < \text{value} \\ 0 & \text{otherwise} \end{cases}$													
value	vector	Value(s) for comparison, if type $\neq$ 'Level0'												
dInd <sub>ASM</sub>	3x3 matrix	Change in CCDB.Bus.Static.Ind <sub>ASM</sub> matrix characterizing magnetically induced stray field vector at ASM sensor position when IndState changes from 0 to 1												
dInd <sub>VFM</sub>	3x3 matrix	Same as dInd <sub>ASM</sub> for CCDB.Body.Ind <sub>VFM</sub> and VFM sensor position												
dAM <sub>ASM</sub>	3x3 matrix	Change in CCDB.AOCS.M <sub>ASM</sub> matrix characterizing direct and induced stray field vector of AOCS magnetorquers at ASM sensor position when IndState changes from 0 to 1												
dAM <sub>VFM</sub>	3x3 matrix	Same as dAM <sub>ASM</sub> for CCDB.AOCS.M <sub>VFM</sub> and VFM sensor position												

Table 5-4 InducedInfo – Elements of CCDB.Bus.Induced Array

#### 5.4.5 Calculation of Magnetorquer Currents

The actual currents in the magneto-torquers,  $Level0.AOCS.I_{mea\_MTQ}$ , are calculated as follows:

$$I_{mea\_MTQ_{i+3v}} = I_i + D(I_i) \tag{5.4-10}$$

$$I_{mea\_MTQ_{i+3(1-v)}} = 0 \tag{5.4-20}$$

where

$$v = L0.Bus.PCDU\_A\_RuALclStat \tag{5.4-30}$$

v = 0 is nominal, v = 1 is redundant.

$$I_i = \frac{CCDB.AOCS.G\_MTQ \quad CCDB.AOCS.VZ\_MTQ}{L0.AOCS.OBC\_MTQ\_ZENER+ - L0.AOCS.OBC\_MTQ\_ZENER- - (L0.AOCS.MTQ_i\_Curr - L0.AOCS.OBC\_MTQ\_ZENER-)} \quad (5.4-40)$$

and

$$D(x) = \begin{cases} CCDB.AOCS.DH\_MTQ & \text{if } x \geq 0 \\ CCDB.AOCS.DL\_MTQ & \text{if } x < 0 \end{cases} \quad (5.4-50)$$

L0.AOCS corresponds to HK\_xNOM\_0\_ (PID=4) SID=6, L0.Bus is SID=7.

#### 5.4.6 Conversion of Cold Gas and Optical Bench HK Data

The AOCS propulsion (cold gas tanks) house-keeping telemetry, Levella.AOCS.Prop\_HK, and the Optical Bench house-keeping telemetry, Levella.Bus.OpticalBench, is computed from HK\_xNOM\_0\_ (PID=4) SID=8 and SID=47 respectively by

$$X = \text{Interpol}_{Lin}(TMC\_ID, L0.ID) \quad (5.4-55)$$

where X is a P\_A, P\_B, or V\_i of Levella.AOCS.Prop\_HK or any of the thermistor voltages, V\_C\_i, which are converted to temperatures, T\_C\_i, as described in [AD01]. TMC\_ID is CCDB look-up table identifier, and ID is the Level0 channel identifier. The relation between X, TMC\_ID, and ID is given in the following table:

X	TMC_ID	ID	Description
CCDB.TMC.SWARM_AuxSystemHkTm8TankTemp.TMC_ID			
P_A	HPT_1_AN1__Signal	AHT00000	Pressure transducer, Tank A
P_B	HPT_2_AN1__Signal	AHT00002	Pressure transducer, Tank B
T_A <sub>1</sub>	TCM_5_ANY_TankA1	THT00083	Temperature 1, Tank A
T_A <sub>2</sub>	TCM_7_ANY_TankA2	THT00113	Temperature 2, Tank A
T_A <sub>3</sub>	TCL_1_ANY_02A_TankA	THT00014	Temperature 3, Tank A
T_A <sub>4</sub>	TCL_3_ANY_02B_TankA	THT00042	Temperature 4, Tank A
T_A <sub>5</sub>	TCL_5_ANY_02C_TankA	THT00070	Temperature 5, Tank A
T_B <sub>1</sub>	TCM_5_ANY_TankB1	THT00084	Temperature 1, Tank B
T_B <sub>2</sub>	TCM_7_ANY_TankB2	THT00114	Temperature 2, Tank B
T_B <sub>3</sub>	TCL_1_ANY_03A_TankA	THT00015	Temperature 3, Tank B
T_B <sub>4</sub>	TCL_3_ANY_03B_TankA	THT00043	Temperature 4, Tank B
T_B <sub>5</sub>	TCL_5_ANY_03C_TankA	THT00071	Temperature 5, Tank B
V_i	OBC__REF_3p3V_FGM_A	OHT00024	Voltage reference

X	TMC_ID	ID	Description
CCDB.TMC.SWARM_AuxSystemHkTm2TTCThermistors.TMC_ID			
T_if <sub>1</sub>	TCL_2_ANG_14A_ObBoomIf	THT00029	Temperature 1, Optical Bench Boom I/F
T_if <sub>2</sub>	TCL_4_ANG_14B_ObBoomIf	THT00057	Temperature 2, Optical Bench Boom I/F
T_if <sub>3</sub>	TCL_6_ANG_14C_ObBoomIf	THT00087	Temperature 2, Optical Bench Boom I/F
T_rad <sub>1</sub>	TCL_2_ANG_17A_ObSubsys	THT00032	Temperature 1, Optical Bench
T_rad <sub>2</sub>	TCL_4_ANG_17B_ObSubsys	THT00060	Temperature 2, Optical Bench
T_rad <sub>3</sub>	TCL_6_ANG_17C_ObSubsys	THT00090	Temperature 3, Optical Bench

Table 5-5 Prop\_HK Conversion Identifiers

The Optical Bench house-keeping telemetry are provided in a 16 s rate.

#### 5.4.7 VFM Data Conversion

##### 5.4.7.1 VFM Science Data

The VFM measurement data are transmitted in one of two ways: compressed (SID=11) or uncompressed (SID=14). If compressed data, 1<sup>st</sup> order differences are used, and

$$Ba_i = \begin{cases} \text{twos\_complement}(Ba_{i-1}^{\text{raw}}) & \text{for } i = 1 \\ Ba_{(i-1)} + \text{twos\_complement}(DBa_i^{\text{raw}}) & \text{for } i = 2, 3, \dots, 50 \end{cases} \quad (5.4-70)$$

where  $Ba^{\text{raw}}/DBa^{\text{raw}}$  is the field component value from Level 0 from axis  $a$ ,  $a = X, Y, Z$ .  $B^{\text{raw}}$  and  $DB^{\text{raw}}$  are 24-bit respectively 8-bit unsigned integers, twos\_complements is the regular two's complement function: if  $x$  is an  $n$ -bit unsigned integer, then

$$\text{twos\_complement}(x) = \begin{cases} x & \text{if } x < 2^{n-1} \\ x - 2^n & \text{otherwise} \end{cases} \quad (5.4-75)$$

If uncompressed data, BX, BY, BZ are stored directly in the ISP as 24-bit 2s-complement.

To convert  $Ba$  to engineering units, i.e. to Level 1a values, divide by  $2^{6.5}$ :

$$\text{Level1a.VFM.EU}_a = Ba / 2^{6.5} \quad (5.4-80)$$

##### 5.4.7.2 VFM Temperatures

The VFM temperature readings are converted using 7<sup>th</sup> order polynomials:

$$\text{Level1a.VFM.T}_{XX} = \sum_{i=0}^7 C_{XX,i} \text{VET}_{XX}^i \quad (5.4-90)$$

where  $XX = CSC, CDC, EU$ ;  $C_{XX,i}$  are coefficients stored in the VFM ICDB; and

$$VET_{XX} = RET_{XX}/2^{16} \cdot 5V + 2.5V \quad (5.4-100)$$

where  $RET_{XX}$  being the 2's-complement signed values (16-bits) of the SCI\_Temp field of the VFM ISP, with  $XX = CSC, CDC, EU$  for  $L0.HK\_Source = 1, 2, 3$  respectively.

#### 5.4.8 ASM Data Conversion

The ASM 1 Hz data (scalar and vector mode) conversions are as follows:

$$Level1a.ASM.E = 10^9 \cdot Level0.ASM.B0 \cdot k\_LSB \quad (5.4-110)$$

where  $k\_LSB$  is stored in the ASM ICDB.

$$Level1a.ASM.T_{Sensor} = \left( A + B \cdot \ln(R) + C \cdot (\ln(R))^3 \right)^{-1} - 273.15 \quad (5.4-120)$$

where  $A, B,$  and  $C$  are parameters in the Temp\_Sens structure of the ASM ICDB, and

$$R = \frac{U \cdot (R_{filter} + R_{ref}) - 2.5 R_{filter}}{2.5 - U} \quad (5.4-130)$$

where  $R_{filter}$  and  $R_{ref}$  are parameters in the Temp\_Sens structure, and

$$U = 2.5 (L0.Temp + 1/2) / (2^{12} - 1) \quad (5.4-140)$$

where  $L0.Temp$  is the raw temperature data of the ASM telemetry.

ASM 250 Hz burst mode data processing producing 1 Hz Level1a.ASM.E data is described in [AD05a].

#### 5.4.9 ACC Data Conversion

See Section 6.2 of [AD13].

#### 5.4.10 STR Data Conversion

In the beginning of the Swarm Mission, the STR instruments suffered from a phenomenon named “false leap seconds” causing individual STR samples to be time-stamped (ca.) one second off. The following ad-hoc routine detects and corrects<sup>3</sup> these. Let  $t_i^{raw}$  be the sequence of STR attitude timestamps of a given CHU, and  $t_i$  the corresponding sequence of corrected timestamps (in units of seconds). Then:

$$t_i = \begin{cases} t_i^{raw} - 0.992 & \text{iff } t_i^{raw} - t_{i-1}^{raw} > 1.5 \text{ and } t_{i+1}^{raw} - t_i^{raw} < 0.5 \\ t_i^{raw} + 1 & \text{iff } t_i^{raw} - t_{i-1}^{raw} < 0.5 \text{ and } t_{i+1}^{raw} - t_i^{raw} > 1.5 \\ t_i^{raw} & \text{otherwise} \end{cases} \quad (5.4-150)$$

The STR temperature telemetry from  $HK\_xNOM\_0_$  (PID=4) SID=46 are provided in degrees celsius at 1 Hz and is stored directly into Level1a.STR.HK.

<sup>3</sup> Consecutive samples suffering from false leap seconds are not corrected.

## 5.5 Level 1a Data Description

The description of Level 1a data elements required for the Level 1b Processor is given in [AD14]. The data are grouped into blocks where each block contains a coherent set of observations. The relation between Level 1a Product Names and Data Sets and corresponding Levella data structure is given in the table below.

Product	Data Set	Levella element
ASMxSCI_1a	MDR_ASM_SCI	ASM.Sci
VFMxSCI_1A	MDR_VFM_SCI	VFM.Sci
STRxSCI_1A	MDR_STR_SCI	STR.Sci
ACCxSCI_1A	MDR_ACC_SCI	ACC.Sci
EFIx_LP_1A	<i>See [AD08]</i>	
EFIxTII_1A	<i>See [AD08]</i>	
GPSxNOM_1A	<i>See [AD11]</i>	
HK_x_AOCS1A	MDR_MAG_HK	AOCS.Mag_HK
	MDR_THRU_HK	AOCS.Thru_HK
	MDR_PROP_HK	AOCS.Prop_HK
HK_xBUS_1A	MDR_STAT_HK	Bus.State
	MDR_CURR_HK	Bus.Curr

## 6 Magnetic and Attitude Data Processing

This section describes the processing of magnetic and attitude data of Swarm. The primary Level 1b magnetic data product elements and ephemeris attitude element are listed Table 6-1.

Product <sup>4</sup>	Element	Frame	Rate	Description
Mag-H	$\mathbf{B}_{\text{VFM}}$	VFM	~50 Hz	Magnetic vector data in the orthogonal VFM instrument frame at the basic VFM instrument sampling rate
Mag-L	$\mathbf{B}_{\text{VFM}}$	VFM	1 Hz	As above but reduced to 1 Hz
Mag-H	$\mathbf{B}_{\text{NEC}}$	NEC	~50 Hz	Magnetic vector data in the NEC (North-East-Center) terrestrial frame at the basic VFM instrument sampling rate
Mag-L	$\mathbf{B}_{\text{NEC}}$	NEC	1 Hz	As Mag-L. $\mathbf{B}_{\text{VFM}}$ above but in NEC frame
Mag-L	F		1 Hz	Magnetic field intensity ( <i>scalar magnetic field</i> ) – provided by the ASM instrument
Mag-H	$\mathbf{q}_{\text{CRF}}$	NEC ← CRF	~50 Hz	Transformation: CRF (STR Common Reference Frame) to NEC frame
Mag-L	$\mathbf{q}_{\text{CRF}}$	NEC ← CRF	1 Hz	Transformation: CRF (STR Common Reference Frame) to NEC frame
Eph	$\mathbf{q}$	ITRF ← S/C	1 Hz	Transformation: S/C to ITRF frame

Table 6-1 Main Level 1b Product Elements of Section 6

Generally, the processing of magnetic data consists of:

- Application of intrinsic instrument calibration parameters
- Subtraction of magnetic disturbances – magnetic stray fields from the spacecraft
- Resampling to desired sampling frequencies and/or time instants (not Mag-H. $\mathbf{B}$  data)
- Translation of vector data into appropriate frame – using among other the attitude data of the STR

The processing of magnetic and attitude data is depicted in Figure 6-1 below with telemetry and some ephemeris inputs at the top and (main) Level 1b outputs at the bottom. The elements are further described in Table 6-2 below the figure. The details of the processing within the boxes are described in the Sections as indicated in the figure. The order of processing is largely dictated by the elements being passed between the individual processing boxes. To

<sup>4</sup> Here, the short Level 1b Product names are used, see Table 4-2 for the “official” names.

ease the description of the processings, the Sections focus on the single instruments, but the overall sequence of the processing is as follows:

1. Process GPSR and STR instrument level data, and perform MOD (Section 8 and 6.4)
2. Process VFM data until  $\mathbf{B}'_{VFM}$  and  $t_{out,VFM}$  are reached.
3. Interpolate attitude data (from Step 1.)
4. Process ASM data generating  $F_{ASM}$ ,  $t_{ASM}$ , and Level 1b.Mag-L.F (including related elements)
5. Finalize VFM processing: estimate and apply TCF.VFM parameters, compute Level 1b.Mag-H. $\mathbf{B}_{VFM}$  and Level 1b.Mag-L. $\mathbf{B}_{VFM}$ , and transform both to NEC frame ( $\mathbf{B}_{NEC}$ ).

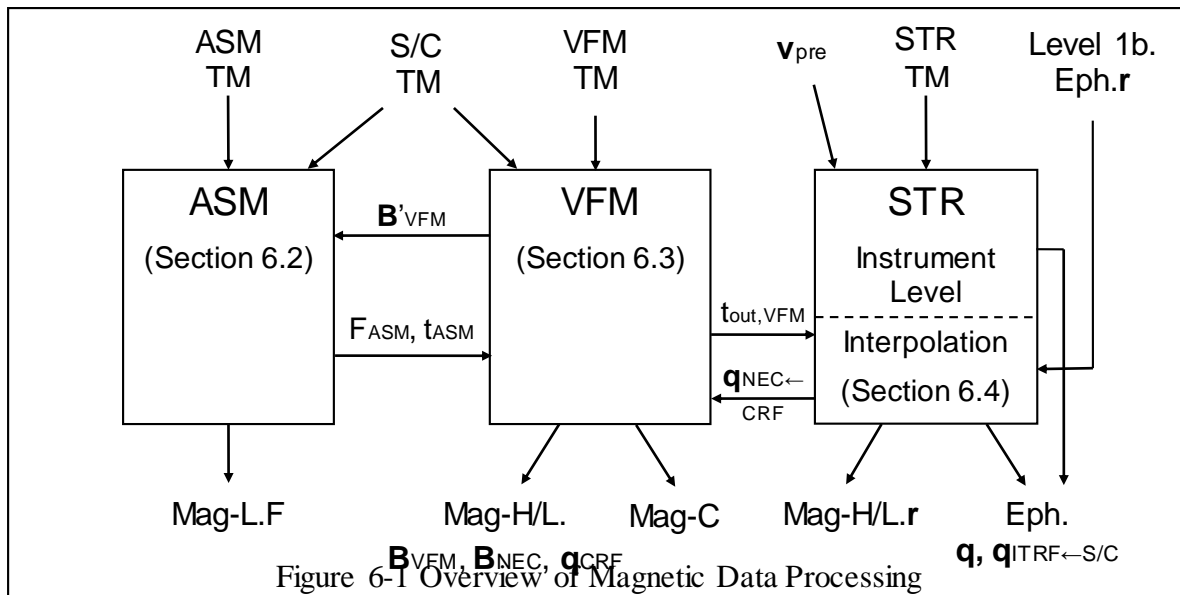


Figure 6-1 Overview of Magnetic Data Processing

	Element	Description
Level 1a inputs	ASM TM	ASM telemetry, measurements and HK
	S/C TM	S/C telemetry required to compute magnetic stray fields
	VFM TM	VFM telemetry, measurements and HK
	STR TM	STR telemetry, measurements and required HK
Intermediary input/output	$v_{pre}$	Preliminary S/C velocity, ITRF, required initially
	Level 1b.Eph.r	S/C position, ITRF, output from MOD – required for the final computation of $\mathbf{r}_{VFM}$
	$\mathbf{B}'_{VFM}$	Intermediary magnetic vector data (50 Hz)
	$t_{out,VFM}$	Time-stamps of VFM measurements – 50 Hz
	$F_{ASM}$	Calibrated and corrected scalar magnetic data

	Element	Description
	$t_{ASM}$	Time instants of $F_{ASM}$ measurements
	$\mathbf{q}_{NEC \leftarrow CRF}$	Quaternion representation of $CRF \rightarrow NEC_{VFM}$ transformation, at UTC seconds (1 Hz) and at $t_{out,VFM}$ (50 Hz)
Final Level 1b output	Mag-L.F	Magnetic scalar data, 1 Hz
	Mag-H/L. $\mathbf{B}_{XXX}$	Magnetic vector data, 50 Hz and 1 Hz, VFM and NEC frame
	Mag-H/L. $\mathbf{q}_{CRF}$	Attitudes, same as $\mathbf{q}_{NEC \leftarrow CRF}$ above
	Mag-C	VFM calibration product, see Section 6.3
	Eph. $\mathbf{q}$	Attitudes, STR Instrument Level 1b data
	Eph. $\mathbf{q}_{ITRF \leftarrow S/C}$	Attitudes, 1 Hz, S/C $\rightarrow$ ITRF transformation
	Mag-H/L. $\mathbf{r}$	Positions for magnetic products (VFM sensor position), ITRF

Table 6-2 Elements of Figure 6-1

There are two additional, auxiliary magnetic products, `ASMXAUX_1B` and `VFMXAUX_1B`, containing detailed stray field information to be used for analysis of instrument and S/C performance and data analysis.

## 6.1 Magnetic Stray Fields

The magnetic stray fields are divided into three categories:

- State dependent (including static contribution)
- Electrically induced (by currents)
- Magnetically induced (through soft magnetic materials)

### 6.1.1 State Dependent Stray Fields

The static and state dependent stray fields are modelled by two vectors, *biases*, one for the VFM and one for the ASM sensor position. The total static stray field from the satellite body with all instruments and subsystems turned off are stored in the `CCDB.Bus.Static.BaseXXX` vectors, where `XXX` is ASM respectively VFM. All operational states of an instrument or subsystem – except thruster activation (see below) – causing a change in its steady magnetic field are registered in the `Level 1a.Bus.State.States` vector, and the corresponding change in the magnetic field vector is stored in `dBXXX` of the corresponding entry in `CCDB.Bus.State.Field`. In case electrical heaters produce stray fields, these must be included in `Level 1a.Bus.State.States`. Stray fields caused by thruster activation are characterized by `CCDB.AOCS.ThrusterXXX`. I.e. the actual state dependent stray field at e.g. the VFM sensor position is given by:



$$\begin{aligned}
 \mathbf{dB}_{\text{State,VFM}} = & \text{CCDB.Bus.Static.Base}_{\text{VFM}} + M_{\text{State,VFM}} \times \text{Level 1a.Bus.State.States} + \\
 & \text{CCDB.AOCS.Thruster}_{\text{VFM}} \times \\
 & \text{State}_{\text{Thruster}}(\text{Level 1a.AOCS.Thru\_HK}, \text{L0.Bus.PCDU\_A\_RuALclStat}, \\
 & \text{CCDB.AOCS.Thruster\_Settling\_Time}, t)
 \end{aligned}
 \tag{6.1-10}$$

where

$$M_{\text{State,XXX}}(:,i) = \text{CCDB.Bus.State.Field}(i).\mathbf{dB\_XXX} \tag{6.1-15}$$

i.e. the  $i$ 'th column of  $M_{\text{State,XXX}}$  is  $\text{CCDB.Bus.State.Field}(i).\mathbf{dB\_XXX}$ .

$\text{State}_{\text{Thruster}}$  maps the thruster activation telemetry into a 0/1 vector for given time instant  $t$ .  $\text{Level1a.AOCS.Thru\_HK}$  contains the three fields  $t$ ,  $\mathbf{dt}$  and  $\text{Valid}$  (see Section 5.5) –  $\mathbf{dt}$  and  $\text{Valid}$  are vector/bitmask of length 12 corresponding to 8 attitude control and 4 orbit control thrusters. Only if  $\text{Valid}_i = 1$ , the contents of  $\mathbf{dt}_i$  is valid and contains the duration of the activation of thruster  $i$  – duration since the time-stamp,  $t$ , of  $\text{Level1a.AOCS.Thru\_HK}$ . Note that  $\mathbf{dt}_i$  may be larger than 1 second, i.e. “extending” into the next  $\text{Level1a.AOCS.Thru\_HK}$  packet, and note that thruster duration may be overridden (e.g. terminated if  $\mathbf{dt}_i = 0$ ) in subsequent packets.  $\text{L0.Bus.PCDU\_A\_RuALclStat}$  (cf Section 5.4.5) indicates whether primary ( $v=0$ ) or redundant ( $v=1$ ) thrusters are activated.  $\text{CCDB.AOCS.Thruster\_Settling\_Time}$  contains the estimated vibration settling time of the S/C following thruster activation I.e.

$$\text{State}_{\text{Thruster},i}(\text{Thru\_HK}, v, \mathbf{st}, t) = \begin{cases} 1 & \text{if } t < \text{Thru\_HK}(j).t + \text{Thru\_HK}(j).\mathbf{dt} + \mathbf{st} \\ & \text{where } j \text{ is the largest } j \text{ for which} \\ & \text{Thru\_HK}(j).\text{Valid}_{i-12v} = 1 \text{ and } t \geq \text{Thru\_HK}(j).t \\ 0 & \text{otherwise} \end{cases} \tag{6.1-16}$$

### 6.1.2 Electrically Induced Stray Fields

The electrically induced stray fields originate from various bus currents, e.g. solar panel array currents, battery charge/discharge currents, and in particular currents in the magneto-torquer coils of the AOCS. The stray fields of the AOCS magnetotorquer coils are described temporally by their transfer function and magnetically by linear interpolation (table look-up) followed by matrix multiplication. These matrices ( $\text{CCDB.AOCS.M}_{\text{XXX}}$ ) include the static and default state magnetically induced contributions from the S/C body. The electrically induced stray fields from other sources (Bus currents) are modelled by smoothing filters followed by look-up tables and matrix multiplication.

The parameters for electrically induced stray fields are stored in the  $\text{CCDB.AOCS}$  and  $\text{CCDB.Bus.Current}$  structures. The relevant fields are listed in Table 6-3.

Field	Type	Description	
CCDB.AOCS parameters			
I_xfer	LUT1d	Current transfer look-up tables. Order: MTQ1_A, MTQ2_A, MTQ3_A, MTQ1_B, MTQ2_B, MTQ3_B	
M <sub>Xxx</sub>	3 x 6 matrix	Output multiplication matrix for sensor Xxx Xxx = VFM, ASM	
Transfer	D	real	Temporal characterization parameters of AOCS coil transfer function.
	wo	real	
	dt	real	Width of transfer region
Transfer <sub>VFM</sub>	<i>Characterization of AOCS coil and VFM instrument transfer functions.</i>		
	dt	vector	Time offsets of transfer regions, 3 elements
	coeff1	polynomial	Cubic polynomial for t in [t <sub>0</sub> +dt <sub>1</sub> ..t <sub>0</sub> +dt <sub>2</sub> ]
	coeff2	polynomial	Cubic polynomial for t in ]t <sub>0</sub> +dt <sub>2</sub> ..t <sub>0</sub> +dt <sub>3</sub> ]
Thruster <sub>Xxx</sub>	3 x 24 matrix	Thruster activation stray field matrix, for Xxx sensor position (see Section 6.1.1). Order: primary attitude 1-8, primary orbit 1-4, redundant attitude 1-8, redundant orbit 1-4.	
Max <sub>ASM</sub>	vector	Maximum AOCS stray field vector at ASM sensor	
Max <sub>VFM</sub>	vector	Maximum AOCS stray field vector at VFM sensor	
CCDB.Bus.Current parameters, 3 groups: Yyy = Bat, SP, other.			
Filter <sub>Yyy</sub>	FIR_FP	Filter averaging parameter. See Appendix E Page 148 and [AD03].	
Yyy(i).I_xfer	LUT1d	Current transfer look-up table	
Yyy(i).dB_Xxx	vector	Current dependent stray field vectors for Xxx sensor position, columns in M <sub>Current,Xxx</sub> matrix. Xxx = ASM, VFM	
Max <sub>ASM</sub>	vector	Maximum bus current stray field vector at ASM sensor	
Max <sub>VFM</sub>	vector	Maximum bus current stray field vector at VFM sensor	

Table 6-3 Structure(s) for Electrically Induced and State Dependent Stray Fields

For Bus currents, the stray field is computed as follows: Given an input vector of length k of telemetry currents,  $Levella.Bus.Curr.I_{group}$ , each element is first converted using the corresponding  $CCDB.Bus.Current.group.I\_xfer$  look-up table (using  $Interpol_{Lin}$ ), then they are filtered according to  $CCDB.Bus.Current.Filter_{group}$  using the Filter function (see Appendix E Page 147f for these functions) yielding  $I^{filt,group}$  – except for the battery current, which is a single current measured three times,  $I^{filt,Bat}$  is hence a weighted average of the three measurements with weights  $CCDB.Bus.Current.wt_{Bat}$ :

$$I^{filt,Bat} = \frac{\sum_i wt_{Bat,i} Filter(Interpol_{Lin}(I_{Bat,i}))}{\sum wt_{Bat}} \quad (6.1-18)$$

In the event of large gaps (longer than  $CCDB.Bus.Current.Filt_{group}.Max\_Gap\_Size$ ) there is no stray field correction for Bus currents, and data are flagged with missing Bus telemetry.

Now, the stray field at location  $X_{xx}$  (ASM, VFM, or Body) is calculated as:

$$dB_{Bus,X_{xx}} = \sum_{group=SP,Bat,other} M_{Current, group,X_{xx}} \times I^{filt,group} \quad (6.1-20)$$

where:

$$M_{Current, group,X_{xx}}(:,i) = CCDB.Bus.Current.group(i).dB\_X_{xx} \quad (6.1-25)$$

and  $group$  is SP, Bat, or other for the respective source groups.

The stray fields of the AOCS torquer coils are computed in much the same way as for the Bus currents. However the temporal characteristics of the AOCS fields are known in detail as these currents is carefully controlled; hence the filtering of the telemetry is replaced by temporal transfer characterisation functions. The basic transfer response is sketched in Figure 6-2, and the corresponding stray field is computed using the  $CCDB.AOCS$  parameters as specified in Eq. (6.1-27a). For the VFM, the transfer function of the VFM instrument needs to be considered, hence two different transfer functions for the currents are computed – one for the ASM and one for the VFM – see Eq. (6.1-27c) and Eq. (6.1-27d) respectively.

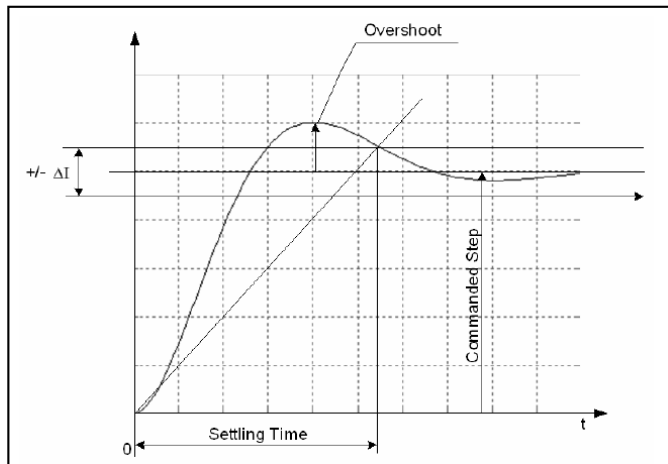


Figure 6-2 AOCS Magneto Torquer Transition

$$dB_{AOCS,X_{xx}}(t) = CCDB.AOCS.M_{X_{xx}} \times I_{AOCS,X_{xx}}(t) \quad (6.1-27a)$$

where  $X_{xx} = ASM$  or  $VFM$ . Given  $I_{AOCS,i} = Interpol_{Lin}(CCDB.AOCS.I\_xfer, Levella.AOCS.Mag\_HK.Imea\_MTQ(i))$ , the  $i^{th}$  torquer current sample measured and transformed at time  $t_{AOCS,i} = Levella.AOCS.Mag\_HK.t(i)$ , then:

$$I_{AOCS,X_{xx}}(t) = I_{AOCS,i} - y_{X_{xx}}(t - t_{AOCS,i}) \mathbf{d}I_{AOCS,i}, \quad (6.1-27b)$$

$$\begin{cases} \text{for } X_{xx} = ASM \text{ and } t_{AOCS,i} \leq t < t_{AOCS,i+1} \\ \text{for } X_{xx} = VFM \text{ and } t_{AOCS,i} \leq t - CCDB.AOCS.Transfer_{VFM} \cdot dt_1 < t_{AOCS,i+1} \end{cases}$$

where  $\mathbf{d}I_{AOCS,i} = I_{AOCS,i} - I_{AOCS,i-1}$ .  $y_{ASM}(dt)$  is given by:

$$y_{ASM}(dt) = \begin{cases} \frac{\sin(\arccos(D) + dt \omega_0 \sqrt{1-D^2})}{e^{dt \omega_0 D} \sqrt{1-D^2}} & \text{for } dt < CCDB.AOCS.Transfer \cdot dt \\ 0 & \text{otherwise} \end{cases} \quad (6.1-27c)$$

where  $D$  and  $\omega_0$  are the parameters of  $CCDB.AOCS.Transfer$ .

$y_{VFM}(dt)$  is given by:

$$y_{VFM}(dt) = \begin{cases} 1 & \text{for } dt < CCDB\dots dt_1 \\ 1 - \text{polyval}(\text{coeff1}, dt) & \text{for } CCDB\dots dt_1 \leq dt \leq CCDB\dots dt_2 \\ 1 - \text{polyval}(\text{coeff2}, dt) & \text{for } CCDB\dots dt_2 < dt \leq CCDB\dots dt_3 \\ 0 & \text{for } dt > CCDB\dots dt_3 \end{cases} \quad (6.1-27d)$$

where  $\text{coeff1}$  and  $\text{coeff2}$  are the parameters of  $CCDB.AOCS.Transfer_{VFM}$ , and  $CCDB\dots dt = CCDB.AOCS.Transfer_{VFM} \cdot dt$  and  $\text{polyval}(P,t)$  is the polynomial specified by  $P$  evaluated at  $t$ .

### 6.1.3 Magnetically Induced Stray Fields

Magnetically induced stray fields are modelled in two parts. One part corresponds to the induced field due to the ambient magnetic field; the other part corresponds to AOCS magnetorquer induced fields in state dependent equipment.

The soft-magnetic field induced by the ambient field is a combination of static+default-state and state-dependent contributions:

$$\mathbf{dB}_{\text{induced},Xxx} = (CCDB.Bus.Static.Ind_{Xxx} + \sum_i \text{Level1a.Bus.State.IndState}_i \text{ CCDB.Bus.State.Induced}_i \cdot dInd_{Xxx}) \times \mathbf{B}'_{\text{Body}} \quad (6.1-30)$$

where  $Xxx$  is ASM or VFM. The sum of the matrices is done element-wise.  $\mathbf{B}'_{\text{Body}}$  is the ambient magnetic field vector,  $\mathbf{B}_{\text{ambient}} = CCDB.Structure.VFM\_q^{-1} \times \mathbf{B}'_{VFM}$ , in the S/C frame computed from the calibrated and corrected VFM measurements (see Section 6.3.4) plus the electrically induced field from bus currents,  $\mathbf{dB}_{\text{Bus,Body}}$  (but not including AOCS currents):

$$\mathbf{B}'_{\text{Body}} = \mathbf{B}_{\text{ambient}} + \mathbf{dB}_{\text{Bus,Body}} \quad (6.1-41)$$

The state dependent, magnetically induced field of the AOCS magnetorquers is computed as follows:

$$\mathbf{dB}_{\text{AOCS-Ind},Xxx} = \left( \sum_i \text{Level1a.Bus.State.IndState}_i \text{ CCDB.Bus.State.Induced}_i \cdot dAM_{Xxx} \right) \times \mathbf{I}_{\text{AOCS},Xxx} \quad (6.1-50)$$

where  $\mathbf{I}_{\text{AOCS},Xxx}$  are the mapped AOCS torquer currents from Eq. (6.1-27b). The summed matrix of Eq. (6.1-50) may be added to  $CCDB.AOCS.M_{Xxx}$  before multiplication with  $\mathbf{I}_{\text{AOCS}}$ :

$$\mathbf{dB}_{\text{AOCS},Xxx} = (CCDB.AOCS.M_{Xxx} + \sum_i \text{Level1a.Bus.State.IndState}_i \text{ CCDB.Bus.State.Induced}_i \cdot dAM_{Xxx}) \times \mathbf{I}_{\text{AOCS},Xxx} \quad (6.1-60)$$

## 6.2 Absolute Scalar Magnetometer – ASM

### 6.2.1 Introduction

The Absolute Scalar Magnetometer (ASM) measures the magnetic field intensity at the tip of the boom. The ASM is an absolute instrument, i.e. it is not subject to changes of its intrinsic parameters over time. It uses the PPS as the absolute, external reference.

[AD05] contains descriptions of the ASM intrinsic processing algorithms for both the nominal scalar mode and the experimental vector mode. Both modes generate scalar (magnetic field intensity) data; the processing of these is described in this chapter, including the *system level* algorithms for stray field correction. Vector data from the ASM vector mode (as well as burst mode data) are only provided as Level 1a data<sup>5</sup>.

### 6.2.2 Input and Output Parameters

Inputs					
Element		Description	Unit	Range <sup>6</sup>	
Level 1a	ASM.Sci.E	Raw ASM measurement	eu		
	ASM.Sci.N	ASM reference frequency count		29 – 30 * 10 <sup>6</sup>	
	AOCS.Mag_HK	Magneto-torquer coil currents	A	±10 <sup>2</sup>	
	AOCS.Thru_HK	Thruster activations		0/1	
	Bus.State	States of instruments/subsystems with varying magnetic disturbance – incl. induced		0/1	
	Bus.Curr	Bus currents e.g. solar panels, main power bus (one vector with all currents)	A	± 2 * 10 <sup>4</sup>	
	<b>B'</b> <sub>VFM</sub>	Calibrated and corrected magnetic field vector measurements from the VFM (using TCF.VFM <sup>init</sup> parameters)	nT	± 66,000	
CCDB	ASMB	b_mod	Vector mode modulation	nT	0 – 10 <sup>3</sup>
		<b>B_rem</b>	Remanent magnetization (static stray field vector)	nT	0 – 10
		B_rf	RF field amplitude	nT	0 – 100
		Delay	Delay between actual measurement and timestamp	s	±1
		Induced	Induced magnetization (matrix)		±10 <sup>4</sup>
		k_BS	Bloch-Siegert correction coefficient	nT <sup>2</sup>	0 – 10 <sup>5</sup>

<sup>5</sup> The ASMXAUX\_1B Level 1b Product contain additional data suited for the Level 1a ASM vector data.

<sup>6</sup> In case of a vector or matrix, **Range** indicates the range of the single elements.

	Filter	Coeff	Matrix holding coefficients of ASM filter designed for 50 Hz sampling frequency		$\pm 5$
		Delay	Delay introduced by ASM filter ( <i>group delay</i> , both scalar and vector mode)	s	0 – 5
L 1 B P . A S M	Dist_Max		Total maximum magnetic disturbance intensity at ASM sensor position	nT	1 – 30
	Freq_Div		NMR frequency divider		$10^2 - 10^3$
	Limits		Structure holding limits for outlier detection and/or rejection, see Table 6-4		
	N_Avg		ASM.Sci.N Filter parameters		
	NMR		He <sup>4</sup> gyromagnetic frequency	GHz/ T	$\approx 28.024$
	Spline_Pars		B-spline parameters for ASM interpolation, see text and [AD03]		
Structure. ASM_q_VFM		Rotation of ASM frame to VFM frame, i. e. VFM→ASM transformation cf. Table 3-1			$\pm 1$
AOCS		Structure characterizing the magnetic stray field vector of the AOCS at ASM sensor position. Includes any non-linearities, time-characteristics of the AOCS coil system, static and default-state induced effects, and the maximum disturbance from the AOCS. See Table 6-3 in Section 6.1.2			
Bus		Structure characterizing the magnetic stray field (electrical and magnetical induced) vector of the S/C; static, state dependent, and bus currents dependent parts. Includes values for the maximum disturbances and uncertainty estimate of stray field calculation. See Table 6-3			
L1BP.VFM.Max_ ASM_VFM_Diff		Maximum ASM – VFM scalar discrepancy, for flagging		nT	0 – 100
VFM.Ind_ASM		Dipole field of VFM at ASM sensor position			$\pm 10^{-4}$
Structure. VFM_q		Rotation of VFM sensor frame to S/C frame			$\pm 1$
<b>Outputs</b>					
<b>Element/Group</b>		<b>Description</b>		<b>Unit</b>	<b>Range</b>
Level 1b <sub>Inst.</sub> -ASM.F		ASM Level 1b data at instrument level, i.e. converted ASM measurements, not corrected for any stray fields		nT	15,000 – 66,000
Level 1b <sub>Inst.</sub> -ASM.F <sub>err</sub>				nT	0 – 100

$t_{ASM}$		Time instants of non-rejected ASM measurements	UTC	
$F_{ASM}$		Fully converted and corrected magnetic field intensity at $t_{ASM}$ time instants (i.e. not corrected for filter group delay)	nT	15,000 – 66,000
Level 1b. Mag-L	F	Fully converted and corrected magnetic field intensity at UTC seconds	nT	15,000 – 66,000
	$F_{error}$	Error estimate of F	nT	0 – 100
	$dF_{AOCS}$	AOCS stray field correction	nT	$\pm 50$
	$dF_{other}$	Stray field correction – all other sources	nT	$\pm 20$
	Flags <sub>F</sub>	Flags related to the ASM data		
	ASM_Freq_Dev	Deviation of frequency calibration data		
ASMXAUX_1B (Level 1b)		Auxiliary Level 1b product containing magnetic stray fields at ASM sensor position		

### 6.2.3 ASM Algorithm Overview

The overall processing is sketched in Figure 6-3 below: First, the raw output (Level 1a.ASM.Sci.E, rate: 1 Hz, timestamps  $t_{0,ASM}$ ) from the ASM is converted to physical units (nT), corrected for the Bloch-Siegert effect, and corrected for delays in timestamps (from  $t_{0,ASM}$  to  $t_{ASM}$ ). This constitutes the Level 1b<sub>Inst</sub>.ASM data, the ASM *instrument Level 1b* scalar product containing the perturbed magnetic field intensity measurements at corrected time instants,  $t_{ASM}$ . Note: this includes stray fields from the ASM itself.

Then outlier detection and/or rejection is performed. Followed by corrections for magnetic disturbances (stray fields) of the ASM itself, of the VFM, and of the rest of the spacecraft. These disturbances need to be filtered according to the intrinsic filter of the ASM instrument (see Section 6.2.4 below). This constitutes the fully calibrated and corrected scalar magnetic measurements,  $F_{ASM}$ , at instrument time instants ( $t_{ASM}$ )

These data are then adjusted for the group delay of the ASM intrinsic filter, i.e. the measurements are shifted in time from  $t_{ASM}$  to  $t_{out,ASM}$ .

Finally, these data are interpolated to yield the scalar elements of the Level 1b.Mag-L product at UTC seconds. The interpolation process also bridges small gaps (a few samples) in the ASM data stream typically caused by missing telemetry (missing ISP) or outlier rejection.

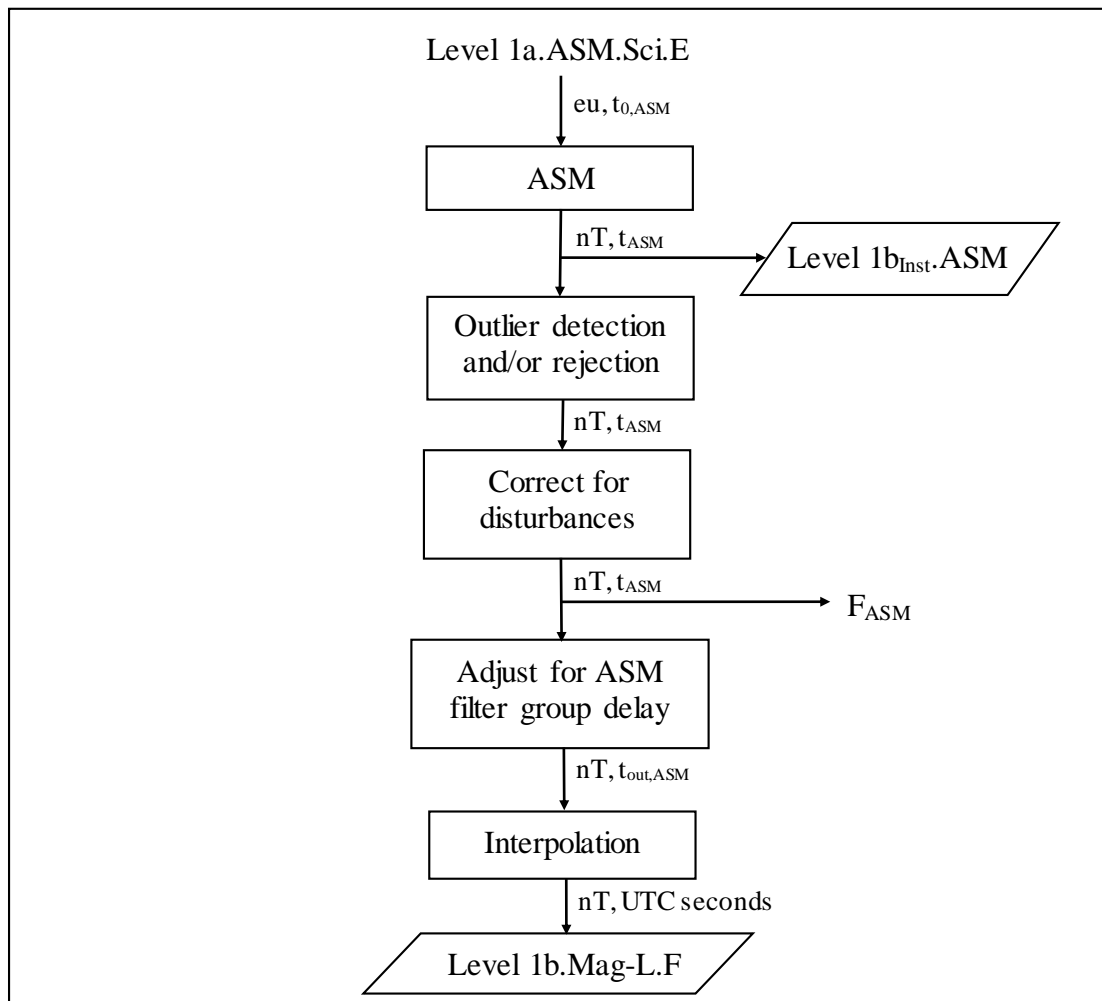


Figure 6-3 ASM Processing Overview

#### 6.2.4 Detailed Description of ASM Processing

The processing of the ASM data is shown in detail in Figure 6-4 on the next page. The elements in the figure are as follows:

$F^{(i)}$	The magnetic field intensity after processing step i. These are given at the corrected time instants of the ASM samples, $t_{ASM}$ , except $F^{(4)}$ which are given at $t_{out,ASM}$ time instants.
$F^{(i)}_{err}$	The uncertainty estimate of $F^{(i)}$ .
$\Delta F_{Total}$	The total magnetic disturbance from the S/C at the ASM sensor location, filtered according to the ASM intrinsic filter.
$\kappa$	Uncertainty of $\Delta F_{Total}$ due to lack of needed telemetry.
$\varepsilon$	Total uncertainty estimate of $\Delta F_{Total}$ .



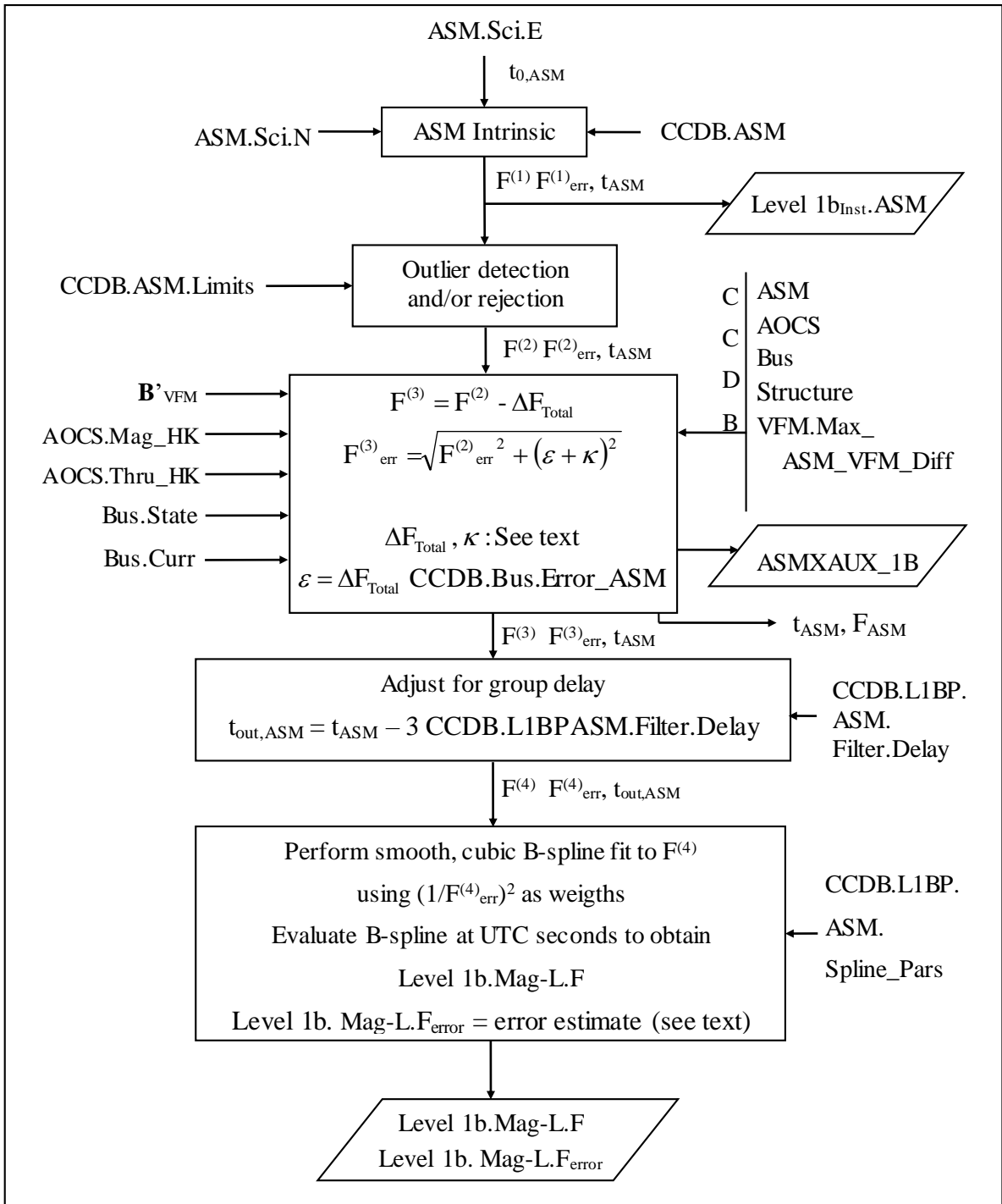


Figure 6-4 ASM Processing

### 6.2.4.1 ASM Intrinsic Processing

The ASM intrinsic processing consists of conversion of the raw measurements (Level 1a. ASM.Sci.E) into physical units (nT) using the actual frequency of the internal ASM reference oscillator given by the Level 1a.ASM.Sci.N telemetry (in the following, the “Level 1a” prefix is omitted). As ASM.Sci.N represents the oscillator count between two consecutive PPS (Pulse Per Second) this is first filtered to avoid spurious PPS’s to propagate to the ASM data. This is done with the Filter function, see page 148 (Appendix E) – gaps are handled as for the Bus currents, see Section 6.1.2. Now, with N\_Avg, Freq\_Div and NMR being elements of the CCDB.L1BP.ASM structure:

$$F^{(0)} = \text{ASM.Sci.E Filter}(N\_Avg, \text{ASM.Sci.N}) \text{ Freq\_Div} / \text{NMR} \quad (6.2-10)$$

$F^{(0)}$  is further corrected for the Bloch-Siegert effect and – in vector mode – for the aliasing effect, see [AD05]. I.e. (CCDB.ASM prefix omitted):

$$\begin{aligned} F^{(1)} &= F^{(0)} - \Delta F_{\text{Bloch-Siegert}} - \Delta F_{\text{aliasing}} \\ &= F^{(0)} + k\_BS B\_rf^2 / F^{(0)} - b\_mod^2 / 2F^{(0)} \end{aligned} \quad (6.2-20)$$

Finally, the Level 0 timestamps are corrected for processing delays (group delays introduced by the ASM instrument filter are corrected later):

$$t_{\text{ASM}} = t_{0,\text{ASM}} - \text{CCDB.ASM.Delay} \quad (6.2-30)$$

The error estimate,  $F^{(1)}_{\text{err}}$  is set to 10 pT.<sup>7</sup>

Falgs<sub>F</sub> is set according to the availability of ASM.Sci.N data for the filter in Eq. (6.2-10), and according to the presence of ASM sampling start or a data gap within 8 seconds prior to the current sample.

The ASM\_Freq\_Dev quality indicator of the ASM instrument is the square root of the deviation of Level1a.ASM.Sci.N and the filtered value of same. Let  $N = \text{Level1a.ASM.Sci.N}$  and  $N^{\text{filt}} = \text{Filter}(\text{CCDB.L1BP.ASM.N\_Avg}, N)$  ( $N^{\text{filt}}$  is used in Eq. (6.2-10)) then:

$$\text{ASM\_Freq\_Dev} = \begin{cases} \sqrt{N - N^{\text{filt}}} & \text{if } N \geq N^{\text{filt}} \\ -\sqrt{N^{\text{filt}} - N} & \text{if } N^{\text{filt}} > N \end{cases} \quad (6.2-35)$$

### 6.2.4.2 Outlier Detection

Outliers are detected by pure out-of-range check plus a range check of 2<sup>nd</sup> order time derivatives of  $F^{(1)}$ .

The CCDB.L1BP.ASM.Limits structure contains the following fields:

Field	Type	Unit	Comment
Range	vector	nT	Allowed range, e.g. [10,000..66,000]
Max_Accel	real	nT/s <sup>2</sup>	Maximum allowed acceleration

<sup>7</sup>  $F^{(1)}_{\text{err}}$  is no longer raised after sampling start or a data gap – these data are only flagged.

Field	Type	Unit	Comment
Pass_Accel	real	nT/s <sup>2</sup>	Accelerations below this are accepted Between Pass_Accel and Max_Accel samples are flagged as <i>suspicious</i> and associated with raised uncertainty estimates
Max_Gap_Size	real	s	Definition of size of data gaps

Table 6-4 CCDB.L1BP.ASM.Limits Structure

First, all  $F^{(1)}$  samples outside Range are discarded. Then, the 1<sup>st</sup> and 2<sup>nd</sup> order time derivatives,  $F'$  respectively  $F''$ , of the remaining field measurements are computed:

$$F'_i = (F_i - F_{i-1}) / (t_i - t_{i-1}) \quad (6.2-40)$$

$$F''_i = 2(F'_{i+1} - F'_i) / (t_{i+1} - t_{i-1}) \quad (6.2-50)$$

where in Eq. (6.2-50)  $i = 2, 3, \dots, n-1$ .

For all blocks of consecutive outliers, i.e. blocks for which  $|F''_i| > \text{Max\_Accel}$  for all  $i = i_1, \dots, i_2$ , remove samples depending on the length of the block as follows:

1. If block has one sample, remove sample  $F_{i_1}$
2. If block has two samples, remove sample with highest  $|F''|$
3. If block has three or more samples, remove samples  $i_1+1, \dots, i_2-1$  (all but first and last)

Additionally, if the block begins or ends at a gap in data or at the ends of the data:

4. If  $(t_{i_1} - t_{i_1-1}) > \text{CCDB.L1BP.ASM.Spline\_Pars.Max\_Gap\_Size}$  remove also sample  $F_{i_1}$
5. If  $(t_{i_1-1} - t_{i_1-2}) > \text{CCDB.L1BP.ASM.Spline\_Pars.Max\_Gap\_Size}$  or  $i_1 = 2$  (at beginning of data) remove also samples  $F_{i_1-1}$  and  $F_{i_1}$
6. If  $(t_{i_2+1} - t_{i_2}) > \text{CCDB.L1BP.ASM.Spline\_Pars.Max\_Gap\_Size}$  remove also sample  $F_{i_2}$
7. If  $(t_{i_2+2} - t_{i_2+1}) > \text{CCDB.L1BP.ASM.Spline\_Pars.Max\_Gap\_Size}$  or  $i_2 = n-1$  (at end of data) remove also samples  $F_{i_2}$  and  $F_{i_2+1}$

When completed, recomputed  $F''$  using non-rejected samples and re-perform the checks above for blocks of  $|F''_i| > \text{Pass\_Accel}$  this time marking samples as suspicious (flag value 4 in Table 6-5) and raised error estimates of all samples  $j$  marked as suspicious:

$$F^{(2)}_{j,\text{err}} = \sqrt{(F^{(1)}_{j,\text{err}})^2 + \Delta F_j^2} \quad (6.2-60)$$

$$\Delta F_j = F_j - (F_{j-1} + (F_{j+1} - F_{j-1}) (t_j - t_{j-1}) / (t_{j+1} - t_{j-1})) \quad (6.2-70)$$

$\Delta F_j$  is the difference between  $F_j$  and the linearly interpolated value of  $F_{j-1}$  and  $F_{j+1}$ .  $F_{j-1}$  and  $F_{j+1}$  are non-outliers but possibly suspicious. If at end of data ( $j = 1$  or  $n$ ) use extrapolation in Eq. (6.2-70).

### 6.2.4.3 Stray Fields

The calculation of stray fields is done at 50 Hz data rate at  $t_{out,VFM}$  time instants in order to obtain sufficient accuracy in the later filter processing;  $t_{out,VFM}$  is chosen for convenience since VFM measurements are used for ambient field information. The stray field vectors are then converted to their scalar equivalents using  $\mathbf{B}'_{VFM}$ , and filtered according to the ASM instrument filter (designed for 50 Hz data rate). Finally, these data are interpolated to the  $t_{ASM}$  time instants and subtracted from the  $F^{(2)}$  measurements.

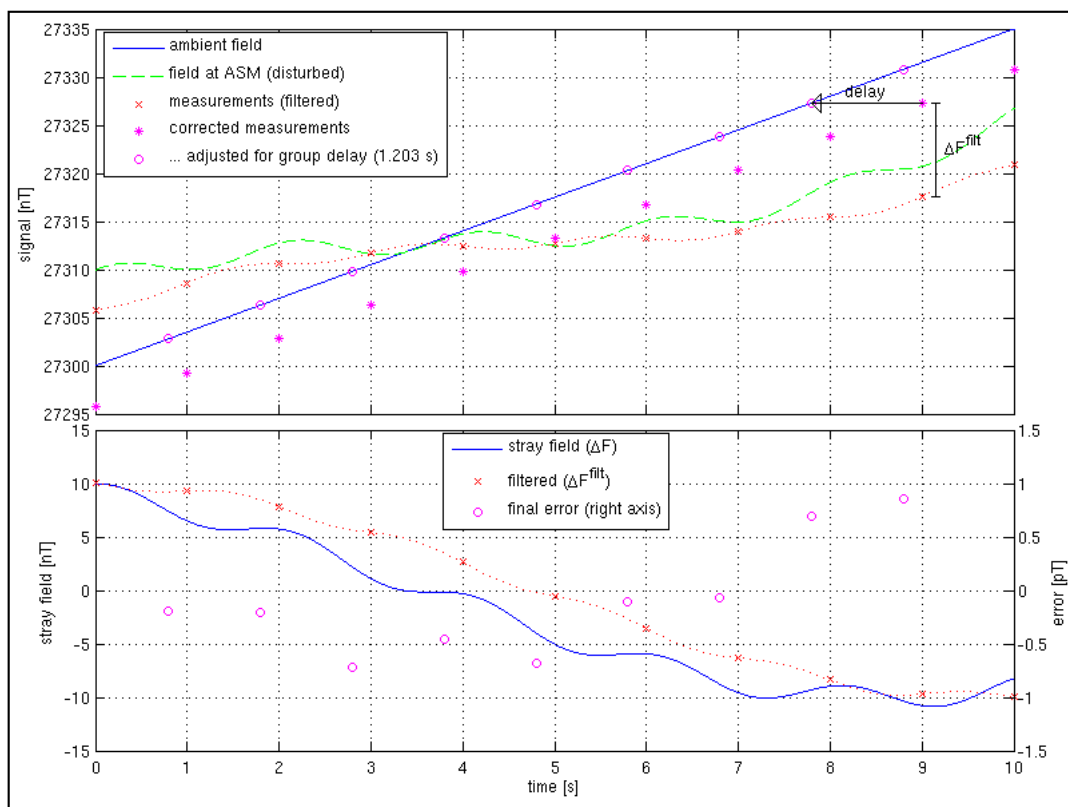


Figure 6-5 Example of ASM Measurements and Stray Fields

Figure 6-5 above illustrates an example of the stray field effect on ASM measurements showing, in the upper plot, both undisturbed (“ambient field” in blue) and disturbed (green) field intensities as well as the corresponding ASM measurements (filtered internally in the ASM at 1 kHz, shown in red). The lower plot shows the stray field intensity (blue) and its filtered value (red). The application of this is indicated in the upper plot by the vertical  $\Delta F^{filt}$  bar and shown by the magenta stars, which are then corrected for the filter group delay to yield the undisturbed field measurements (magenta circles). The final errors between the measurements and the true field are shown to be virtually zero ( $< 1$  pT) by magenta circles in the lower plot.

The processing required for the stray field computation is shown in Figure 6-6 below. First, any gaps in the  $\mathbf{B}'_{VFM}$  data less than 60 seconds in length are filled with linearly interpolated

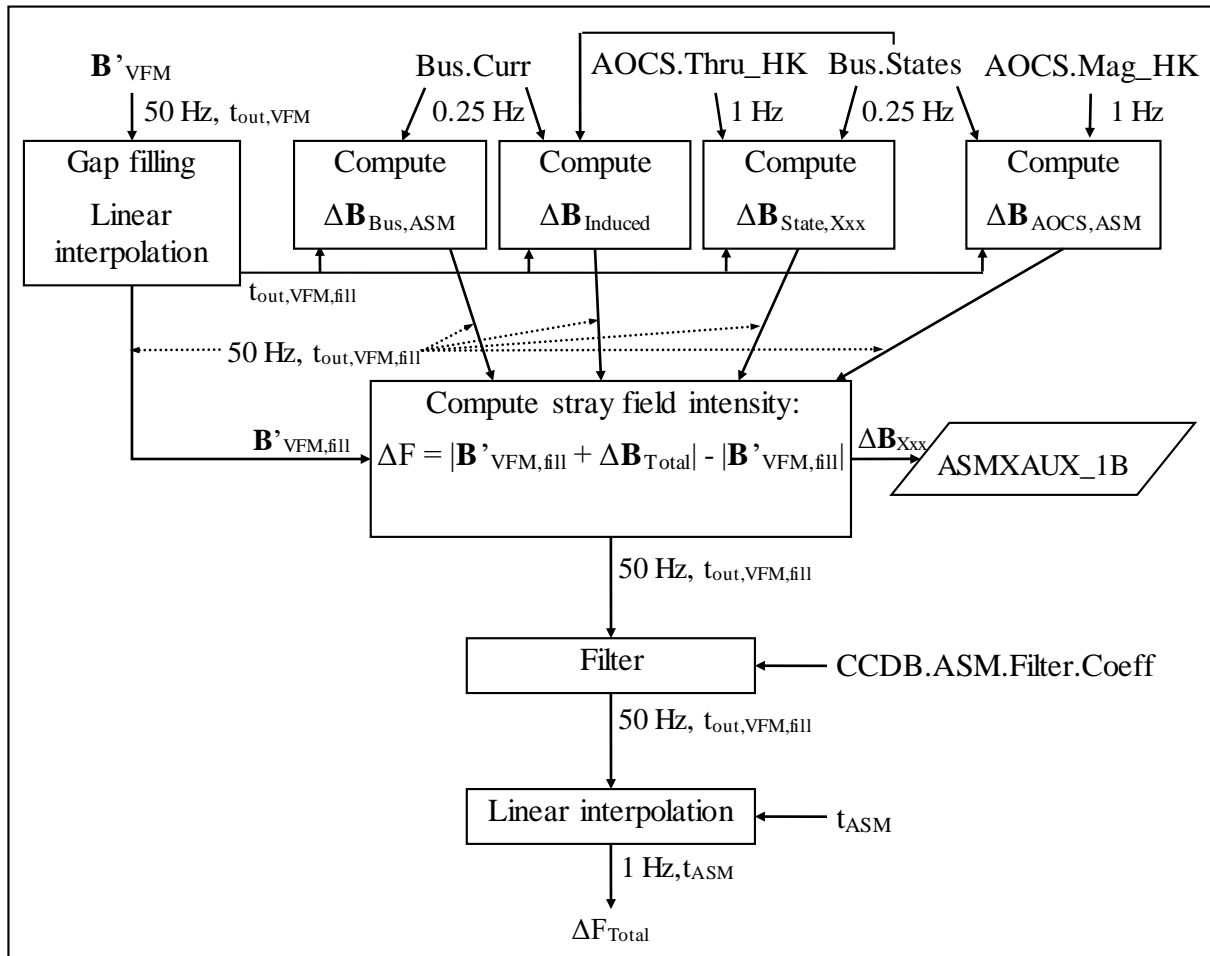


Figure 6-6 ASM Stray Field Processing

values. This also provides the  $t_{out,VFM,fill}$  time-instants. Then, the  $Bus.Curr.I_{group}$  telemetry currents are filtered, linearly interpolated to the  $t_{out,VFM,fill}$  time-instants, and converted to stray field vectors ( $\Delta \mathbf{B}_{Bus,XXX}$ ) using the  $CCDB.Bus$  specification (see Section 6.1.2). This must be done for the ASM sensor ( $XXX = ASM$ ) as well as S/C body ( $XXX = Body$ ) which is used in the calculation of the magnetically induced field. The  $AOC.S.Mag\_HK.I_{mea\_MTQ}$  currents are transformed according to the  $CCDB.AOCS$  (which includes the temporal properties of the transfer function of the AOCS) and  $CCDB.Bus.State.Induced.dAM_{ASM}$  specifications to yield  $\Delta \mathbf{B}_{AOCS,ASM}$  at the  $t_{out,VFM,fill}$  time-instants, cf Section 6.1.2. Note: the  $\Delta \mathbf{B}_{AOCS,ASM}$  only contains the stray fields of the magneto-torquer coils, it does not include stray fields from latch valves and thrusters – these are contained in  $\Delta \mathbf{B}_{Bus,ASM}$ .

$\Delta \mathbf{B}_{State,ASM}$  is computed (cf. Section 6.1.1):

$$\Delta \mathbf{B}_{State,ASM} = CCDB.Bus.Static.Base_{ASM} + M_{State,ASM} \times Interpolate_{Step}(Levella.Bus.State.States, t_{out,VFM,fill}) + CCDB.AOCS.Thruster_{XXX} \times State_{Thruster}(Levella.AOCS.Thru\_HK, t_{out,VFM,fill}) \quad (6.2-80)$$

$Interpolate_{Step}$  simply repeats the last state at the provided input time instants ( $t_{out,VFM,fill}$ ).  $State_{Thruster}$  returns the thrusters states at the provided input time instants ( $t_{out,VFM,fill}$ ).

The magnetically induced stray field contribution is computed (cf. Section 6.1.3):

$$\Delta \mathbf{B}_{\text{Induced}} = (\text{CCDB.Bus.Static.Ind}_{\text{ASM}} + \sum_i \text{Level1a.Bus.State.IndState}_i \cdot \text{CCDB.Bus.State.Induced}_i \cdot \text{dInd}_{\text{ASM}}) \times (\mathbf{VFM\_q}^{-1} \times \mathbf{B}'_{\text{VFM,fill}} + \Delta \mathbf{B}_{\text{Bus,Body}}) \quad (6.2-85)$$

The CCDB.Structure prefix of  $\mathbf{VFM\_q}$  (and  $\mathbf{ASM\_q\_VFM}$  below) has been omitted.

The next step computes the magnetic stray field from the VFM and ASM,  $\Delta \mathbf{B}_{\text{VFM}}$  respectively  $\Delta \mathbf{B}_{\text{ASM}}$ , and adds all the stray field parts into  $\Delta \mathbf{B}_{\text{Total}}$ . Note:  $\Delta \mathbf{B}_{\text{VFM}}$ ,  $\Delta \mathbf{B}_{\text{ASM}}$  and  $\Delta \mathbf{B}_{\text{Total}}$  are in the VFM frame, all the other  $\Delta \mathbf{B}_{\text{xxx}}$  are in the S/C frame.

$$\Delta \mathbf{B}_{\text{VFM}} = \text{CCDB.VFM.Ind\_ASM} \times \mathbf{B}'_{\text{VFM,fill}} \quad (6.2-90)$$

The direct, un-filtered Sun-induced stray field at the ASM is computed via linear 2-D lookup:

$$\Delta \mathbf{B}_{\text{Sun,ASM}}^{\text{look-up}} = \text{Interpol}_{\text{Lin}}(\alpha, \beta, \text{CCDB.L1BP.ASM.dB\_Sun\_ASM}) \quad (6.2-95)$$

This quantity is then filtered

$$\Delta \mathbf{B}_{\text{Sun,ASM}} = \sum_{i=1}^n (b_i \Delta \mathbf{B}_{\text{Sun,ASM}}^{\text{look-up}}(t - i + 1)) - \sum_{i=2}^n (a_i \Delta \mathbf{B}_{\text{Sun,ASM}}(t - i + 1)) \quad (6.2-97)$$

where  $a$  and  $b$  are filter-coefficient vectors of length  $n$  stored in  $\text{CCDB.L1BP.ASM.dB\_Sun\_Filter\_ASM}$ .

$$\Delta \mathbf{B}_{\text{ASM}} = \mathbf{ASM\_q\_VFM}^{-1} \times (\Delta \mathbf{B}_{\text{Sun,ASM}} + \text{CCDB.ASM.B\_rem} + \text{CCDB.ASM.Induced} \times \mathbf{ASM\_q\_VFM} \times \mathbf{B}'_{\text{VFM,fill}}) \quad (6.2-100)$$

$$\Delta \mathbf{B}_{\text{Total}} = \Delta \mathbf{B}_{\text{VFM}} + \Delta \mathbf{B}_{\text{ASM}} + \mathbf{VFM\_q} \times (\Delta \mathbf{B}_{\text{State,ASM}} + \Delta \mathbf{B}_{\text{Bus,ASM}} + \Delta \mathbf{B}_{\text{AOCS,ASM}} + \Delta \mathbf{B}_{\text{Induced}}) \quad (6.2-110)$$

The various  $\Delta \mathbf{B}_{\text{xxx,ASM}}$  elements –  $\Delta \mathbf{B}_{\text{Bus,ASM}}$  and  $\Delta \mathbf{B}_{\text{State,ASM}}$  split into distinct parts – are transformed into the ASM sensor frame by  $(\text{CCDB.Structure,})\mathbf{ASM\_q\_VFM}$  ( $\Delta \mathbf{B}_{\text{VFM}}$ ) or  $\mathbf{VFM\_q} \otimes \mathbf{ASM\_q\_VFM}$  (all other  $\Delta \mathbf{B}_{\text{xxx,ASM}}$ ) and stored in the  $\text{ASMXAUX\_1B}$  Level 1b Product according to the following table, where  $\mathbf{q} = \mathbf{VFM\_q} \times \mathbf{ASM\_q\_VFM}$ , and  $\Delta \mathbf{B}_{\text{Src,ASM}|_{\text{xxx}}}$  indicates the contribution from  $\text{xxx}$  to  $\Delta \mathbf{B}_{\text{Src,ASM}}$ .

ASMXAUX_1B element	Value
t	$t_{\text{out,VFM,fill}}$
$\text{dB}_{\text{Sun}}$	$\Delta \mathbf{B}_{\text{Sun,ASM}}$
$\text{dB}_{\text{AOCS}}$	$\mathbf{q} \times \Delta \mathbf{B}_{\text{AOCS,ASM}}$
$\text{dB}_{\text{Thrust}}$	$\mathbf{q} \times \Delta \mathbf{B}_{\text{State,ASM} _{\text{Thruster}}}$
$\text{dB}_{\text{Battery}}$	$\mathbf{q} \times \Delta \mathbf{B}_{\text{Bus,ASM} _{\text{Battery}}}$
$\text{dB}_{\text{SP}}$	$\mathbf{q} \times \Delta \mathbf{B}_{\text{Bus,ASM} _{\text{Solar Panels}}}$
$\text{dB}_{\text{Bus}}$	$\mathbf{q} \times \Delta \mathbf{B}_{\text{Bus,ASM} _{\text{other}}}$

ASMxAUX_1B element	Value
$\mathbf{dB}_{VFM}$	$\mathbf{ASM\_q\_VFM} \times \Delta \mathbf{B}_{VFM,ASM}$
$\mathbf{dB}_{Static}$	$\mathbf{q} \times \mathbf{CCDB.Bus.Static.Base}_{ASM}$
$\mathbf{dB}_{Ind}$	$\mathbf{q} \times \Delta \mathbf{B}_{Induced(,ASM)}$
$\mathbf{dB}_{State}$	$\mathbf{q} \times \Delta \mathbf{B}_{State,ASM}$ (excl. thrusters and static part)

Now,  $\Delta F$  is calculated:

$$\Delta F = |\mathbf{B}'_{VFM,fill} + \Delta \mathbf{B}_{Total}| - |\mathbf{B}'_{VFM,fill}| \quad (6.2-120)$$

The next step is the filtering of  $\Delta F$ . Let  $\mathbf{a}$  and  $\mathbf{b}$  be the rows of  $\mathbf{CCDB.ASM.Filter.Coeff}$ :

$$\mathbf{b}_i = \mathbf{CCDB.ASM.Filter.Coeff}_{1,i}$$

$$\mathbf{a}_i = \mathbf{CCDB.ASM.Filter.Coeff}_{2,i}$$

Let  $j$  be the sample number ( $j = 1, 2, \dots$ ),  $i$  the filter *depth* ( $i = 0,1,2,3$ ), and  $\Delta F^{(0)} = \Delta F$ , then the filter calculation is as follows:

$$\Delta F_j^{(i+1)} = \mathbf{b}_1 \Delta F_j^{(i)} + \mathbf{b}_2 \Delta F_{j-1}^{(i)} + \mathbf{b}_3 \Delta F_{j-2}^{(i)} - \mathbf{a}_2 \Delta F_{j-1}^{(i+1)} - \mathbf{a}_3 \Delta F_{j-2}^{(i+1)} \quad (6.2-130)$$

( $\mathbf{a}_1 = 1$ , per definition, and is not used.)

The starting condition of the filter is:

$$\Delta F_0^{(i)} = \Delta F_1^{(i)} - \delta^{(i)} \quad \Delta F_{-1}^{(i)} = \Delta F_0^{(i)} - \delta^{(i)}$$

$$\Delta F_0^{(i+1)} = \Delta F_0^{(i)} - \mathbf{d} \cdot \delta^{(i)} \quad \Delta F_{-1}^{(i+1)} = \Delta F_0^{(i+1)} - \delta^{(i)} \quad (6.2-140)$$

$$\delta^{(i)} = \Delta F_2^{(i)} - \Delta F_1^{(i)} \quad \mathbf{d} = \mathbf{CCDB.L1BP.ASM.Filter.Delay} / (t_2 - t_1)$$

$t_1$  and  $t_2$  are the times of  $\Delta F_1$  and  $\Delta F_2$ .

Whenever there is a gap in  $t_{out,VFM,fill}$ , the filter is “re-started”.

The ASM uses three invocations of the filter, hence  $\Delta F^{(3)}$  are the appropriate filtered stray fields. They are linearly interpolated to the  $t_{ASM}$  time instants to yield the  $\Delta F_{Total}$ .

The computation of the recordings of the AOCS stray field corrections in the Level 1b.Mag-C and Mag-L products,  $dF_{AOCS}$ , is done by performing Eq. (6.2-120), the filtering and the interpolation in time using  $\Delta \mathbf{B}_{AOCS}$  in place of  $\Delta \mathbf{B}_{Total}$ . The interpolation in time is done to time instants  $t_{ASM}$  for Mag-C, and to  $t_{UTC} + 3 \mathbf{CCDB.ASM.Filter.Delay}$  for Mag-L (for the record at time  $t_{UTC}$ ). The Sun induced magnetic disturbance corrections in the Level 1b.Mag-C and Mag-L products,  $dF_{Sun}$ , is computed like  $dF_{AOCS}$  using  $\mathbf{ASM\_q\_VFM}^{-1} \times \Delta \mathbf{B}_{Sun,ASM}$  in place of  $\Delta \mathbf{B}_{Total}$ .  $dF_{other}$  is then computed by:

$$dF_{other} = \Delta F_{Total} - dF_{AOCS} - dF_{Sun} \quad (6.2-145)$$

The factor  $\kappa$ , the uncertainty of  $\Delta F_{Total}$  due to lack of telemetry, is computed as the average of the maximum amplitudes of unknown stray fields over the last  $dt = 8 \mathbf{CCDB.ASM.Filter.Delay}$  period. I.e. at time  $t \in t_{ASM}$ :

$$\kappa = \begin{cases} \text{CCDB.L1BP.ASM.Dist\_Max} & \text{if there are gaps in } \mathbf{B}'_{\text{VFM,fill}} \\ & \text{in the time interval [t-dt...t]} \\ \sum_k v_k / n & \text{otherwise. The sum is for all } v_k \\ & \text{in the time interval [t-dt...t],} \\ & \text{n is the number of such } v_k \text{'s} \end{cases} \quad (6.2-150)$$

where  $v_k$  is defined for the time instants of  $\mathbf{B}'_{\text{VFM,fill}}$ ,  $t'_{\text{VFM,fill}}$  as:

$$v_k = v_{\text{Sun,k}} + v_{\text{AOCS,k}} + v_{\text{State,k}} + v_{\text{Curr,k}} \quad (6.2-152)$$

with

$$v_{\text{Sun,k}} = \begin{cases} 0 & \text{if } |t'_{\text{VFM,fill,k}} - t_{\text{Pos,r}}| \leq 60\text{s and} \\ & |t'_{\text{VFM,fill,k}} - t_{\text{Att,q}}| \leq 10\text{s for some r,q} \\ \text{CCDB.L1BP.ASM.dF\_Sun\_Max} & \text{otherwise} \end{cases} \quad (6.2-153)$$

where  $t_{\text{Pos}}$  and  $t_{\text{Att}}$  are the time-instants of the position respectively attitude information nearest in time to  $t'_{\text{VFM,fill}}$

$$v_{\text{AOCS,k}} = \begin{cases} 0 & \text{if } t_{\text{AOCS,r}} \leq t'_{\text{VFM,fill,k}} \leq t_{\text{AOCS,r}} + 1\text{s for some r} \\ \text{CCDB.AOCS.Max}_{\text{ASM}} & \text{otherwise} \end{cases} \quad (6.2-154)$$

$$v_{\text{State,k}} = \begin{cases} 0 & \text{if } t_{\text{State,r}} \leq t'_{\text{VFM,fill,k}} \leq t_{\text{State,r}} + 4\text{s for some r} \\ \text{CCDB.Bus.State.Max}_{\text{ASM}} & \text{otherwise} \end{cases} \quad (6.2-156)$$

$$v_{\text{Curr,k}} = \begin{cases} 0 & \text{if } t_{\text{Curr,r}} - 2\text{s} \leq t'_{\text{VFM,fill,k}} \leq t_{\text{Curr,r}} + 2\text{s for some r} \\ \text{CCDB.Bus.Current.Max}_{\text{ASM}} & \text{otherwise} \end{cases} \quad (6.2-158)$$

where  $t_{\text{xxx}}$  are the time instants of the available Xxx telemetry.

The  $F_{\text{ASM}}$  output is the set of  $F^{(3)}$  ( $= F^{(2)} - \Delta F_{\text{Total}}$ ), these correspond to the  $t_{\text{ASM}}$  time instants.

#### 6.2.4.4 Filter Group Delay Correction

The filter group delay correction is performed through a static time shift – cf. the vertical “delay” arrow in Figure 6-5:

$$t_{\text{out,ASM}} = t_{\text{ASM}} - 3 \text{ CCDB.ASM.Filter.Delay} \quad (6.2-170)$$

The error estimate is propagated as-is:  $F^{(4)}_{\text{err}} = F^{(3)}_{\text{err}}$ .

#### 6.2.4.5 Cubic B-Spline Interpolation

The final interpolation to obtain Level 1b.Mag-L.F at UTC seconds is an iteratively reweighted least squares cubic B-spline approximation to  $F^{(4)}$ . Observe the introduction of leap seconds, cf. Section 4.6. First, the  $F^{(4)}$  data sequence is divided into data segments with data gaps not exceeding the maximum gap size as specified in `CCDB.L1BP.ASM.Spline_Pars.Max_Gap_Size`. The estimation of the B-spline functions to each data segment is susceptible to large deviations at the end points; this effect vanishes if the values at the end points are zero. This is accomplished by linear levelling of each data sequence by  $F^{(4)} - F_{\text{Lin}}$ , where  $F_{\text{Lin}}$  for a given data sequence  $y_0, y_1, \dots, y_{n-1}$  with corresponding time instants  $t_0, t_1, \dots, t_{n-1}$ , is found by:



$$F_{Lin} = \text{interpolate}_{Lin}([t_0, t_{n-1}], [y_0, y_{n-1}], t_i) \quad (6.2-175)$$

Within each leveled segment,  $F^{(4)} - F_{Lin}$ , a cubic B-spline  $f$  is estimated by an iteratively, reweighted least-squares fit that minimizes:

$$\chi^2 = \mathbf{d}^T \mathbf{C} \mathbf{d} + \lambda \mathbf{m}^T \Lambda \mathbf{m} \quad (6.2-180)$$

where:

$\mathbf{m}$  is the *model* vector, i.e. the parameters (spline coefficients) describing  $f$

$\mathbf{d} = (F^{(4)} - F_{Lin}) - f$  is the data residual vector of the current model

$\mathbf{C}$ :  $C_{ij} = w_i G_{ij}$ , where  $G$  is the design matrix (collocation matrix) for cubic B-splines for the given knots (internal spacing CCDB.L1BP.ASM.Spline\_Pars.Knot\_Space, with additional knots in the ends), see Appendix D, and  $\mathbf{w}$  are data weights computed as a combination of Huber weights [RD6],  $\mathbf{w}^{Huber}$ , and the data uncertainties,  $F^{(4)}_{err}$ . Let  $\alpha = \text{Spline\_Pars.Weight\_Huber}$ ,  $\beta = \text{CCDB.L1BP.ASM.Spline\_Pars.Weight\_ErrEst}$ , and  $\sigma_{Ferr} = \text{rms}(F^{(4)}_{err})$  (summed over data within the current segment), then:

$$\mathbf{w} = \alpha \mathbf{w}^{Huber} + \beta \mathbf{w}^{data} + \min(\mathbf{1} - \alpha - \beta, \mathbf{0}) \quad (6.2-190)$$

$\mathbf{w}^{Huber}$  are the Huber weights, they down-weight outliers and are determined iteratively, starting with  $w_i^{Huber} = 1$  in the first iteration.  $\mathbf{w}^{Huber}$  of the next iteration is computed by:

$$w_i^{Huber} = \min\left(1, \frac{1.5}{|d_i/\sigma|}\right) \quad (6.2-200)$$

where  $\sigma$  is the root-mean-squared (rms) misfit of  $\mathbf{d}$  weighted by the weights  $\mathbf{w}$  of the preceding iteration:

$$\sigma = \sqrt{\frac{\sum (w_i d_i)^2}{\sum w_i}} \quad (6.2-210)$$

$\mathbf{w}^{data}$  are the data uncertainty weights;  $\mathbf{w}^{data} = \mathbf{w}^d / \text{mean}(\mathbf{w}^d)$  where  $w^d = 1/(F_{err}^{(4)} + 0.003)$ .

$\lambda$  is the damping factor of the second time-derivative, i.e. the damping of the acceleration of  $f$ ,  $\lambda$  is given in CCDB.L1BP.ASM.Spline\_Pars.Acc\_Damping.

$\Lambda$  is the block diagonal acceleration (second time-derivative) damping matrix, see Appendix D.

$\chi^2$  of Eq. (6.2-180) is minimized by iteratively computing  $\mathbf{m} = (\mathbf{G}^T \mathbf{C} + \lambda \Lambda)^{-1} \mathbf{C}^T (F^{(4)} - F_{Lin})$  and  $\mathbf{w}^{Huber}$  until the relative change in  $\mathbf{m}$  is infinitesimal (though maximum 30 iterations):

$$\sqrt{\sum (m_i - m_i^{prev})^2} / \sqrt{\sum m_i^2} < 10^{-5} \quad (6.2-220)$$

For all UTC seconds within the selected data segment, the final unleveled B-spline,  $f + F_{Lin}$ , is evaluated to yield Level 1b. Mag-L.F.

The error estimate of Level 1b.Mag-L.F, Level 1b.Mag-L.F<sub>error</sub>, is the weighted sum of the error estimate of the four nearest samples (indices  $j$ ):

$$F_{\text{error}} = \beta \frac{\sum F_{j,\text{err}}^{(5)} \text{wt}_j}{\sum \text{wt}_j} \quad (6.2-230)$$

where

$$F_{\text{err}}^{(5)} = \sqrt{\left( (F_{\text{err}}^{(4)})^2 + L_{\text{err}} (F^{(4)} - F_{\text{Lin}} - f)^2 \right) / (1 + L_{\text{err}})} \quad F_{\text{err}}^{(5)} = \sqrt{\left( (F_{\text{err}}^{(4)})^2 + L_{\text{err}} (F^{(4)} - f)^2 \right) / (1 + L_{\text{err}})} \quad (6.2-240)$$

$f$  is the B-splines evaluated at  $t_{\text{out,VFM}}$  time instants,  $L_{\text{err}}$  is given in CCDB.L1BP.ASM.Spline\_Pars.L\_error, and

$$\text{wt}_j = 1 / (|t - t_j| + 10^{-3}) \quad (6.2-250)$$

with  $t$  = time of  $F_{\text{error}}$  and  $t_j$  = time of  $F_{j,\text{err}}^{(5)}$ .

$\beta$  is the uncertainty increase factor due to missing ASM samples – 5 is approximately the minimum, nominal value of  $\sum \text{wt}_j$ :

$$\beta = \max(1, 5 / \sum \text{wt}_j) \quad (6.2-260)$$

#### 6.2.4.6 Flags

The flags of the ASM measurements, Level 1b.Mag-L.Flags<sub>F</sub>, are computed as the sum of the values specified in Table 6-5 below for which the associated conditions are true.

Value	Condition
1	ASM is running in vector mode
2	ASM frequency calibration data trouble: outlier detected, gap, or not enough samples for filtering (by Filter(CCDB.L1BP.ASM.N_Avg, Level 1a.ASM.Sci.N))
4	One or more of the 4 nearest ASM samples is detected as suspicious
8	Within 8 seconds after: <ul style="list-style-type: none"> <li>• ASM restart</li> <li>• Loss of magnetic field lock, or</li> <li>• Telemetry gap</li> </ul>
16	One or more of the 4 nearest ASM samples differs from the VFM measurement, i.e. $ F^{(4)} - \text{InterpolateLin}( \mathbf{B}'_{\text{VFM,fill}} )  > \text{CCDB.Max\_ASM\_VFM\_Diff}$ for at least one of the four nearest $F^{(4)}$ .
32	One or more of the 4 nearest ASM samples are missing or rejected
64	VFM instrument is turned off (i.e. no $\mathbf{B}'_{\text{VFM,fill}}$ data available)
255	Not enough ASM samples available for generating Mag-L.F value

Table 6-5 Values and Conditions of Flags<sub>F</sub>

## 6.3 Vector Field Magnetometer – VFM

### 6.3.1 Introduction

The Vector Field Magnetometer (VFM) measures the magnetic field vector at the tip of the optical bench on the boom. The sensor is a 3-axis Compact Spherical Coil (CSC) with a 3-axis Compact Detector Coil (CDC) inside. The instrument operates as a closed-loop system adjusting the compensating CSC currents to maintain a null field at the detector coils within the sphere. The currents in the CSC coils are measured and digitized (by an ADC) and this constitutes the raw measurements of the instrument. See [AD06] for a detailed description of the instrument, and for the Level 1b algorithm described by the instrument supplier.

The VFM is an analogue instrument and as such subject to temporal changes due to radiation and aging effects of the electronics. The effects are only significant in the bias and linear scale parameters of the characterization; hence these parameters are estimated daily through a comparison of the VFM output with the ASM scalar magnetometer output. Additionally, the non-orthogonality angles of the VFM sensor may also be estimated in this process. The allowed change from day to day in the parameters is controlled group-wise by weight parameters specified in the CCDB. The parameters estimated daily are stored in a *Temporal Calibration File* (TCF) as an auxiliary data product.

### 6.3.2 Input and Output Parameters

Inputs				
Element		Description	Unit	Range
L e v e l 1 a	VFM.Sci.EU	Raw VFM measurement	eu	$\pm 92,682$
	VFM.Sci.T <sub>X</sub>	VFM temperature, multiplexed CDC, CSC, and EU	°C	-40 – +50
	VFM.Sci.Src <sub>T<sub>X</sub></sub>	Temperature source		0 – 3
	VFM.Sci.DPU_id	Active unit identifier		2, 3
	AOCS.Mag_HK	Magneto-torquer coil currents	A	$\pm 0.2$
	AOCS.Thru_HK	Thruster activations		0/1
	Bus.State	States of instruments/subsystems with varying magnetic disturbance		0/1
	Bus.Curr	Bus currents e.g. solar panels, main power bus	A	$\pm 20$
	t <sub>ASM</sub>	Time-instants of screened ASM measurements, not corrected for ASM filter delay	UTC	
	F <sub>ASM</sub>	Calibrated and corrected ASM measurements at t <sub>ASM</sub> time-instants	nT	15,000 – 66,000
	q <sub>NEC←CRF</sub>	Rotation of NEC <sub>VFM</sub> to CRF frame		$\pm 1$
	q <sub>error</sub>	Error estimate of q <sub>NEC←CRF</sub>	"	0 – 10 <sup>3</sup>

C C D B	V F M	C a l	<b>b<sub>0</sub></b>	Bias vector, at 0 °C and t = t <sub>0</sub>	eu	±100
			<b>b<sub>t</sub></b>	Bias vector change with time	eu/yr	±10
			<b>b<sub>EU</sub></b>	Bias vector change with electronics (EU) temperature	eu/°C	±1
			<b>b<sub>CDC</sub></b>	Bias vector change with sensor detector temperature	eu/°C	±1
			<b>deltab<sub>CDC</sub></b>	Bias vector look-up table (Lut1d) for sensor detector temperature	eu	±1
			<b>Filt<sub>T,CDC</sub></b>	Filter parameter for T <sub>CDC</sub>		
			<b>Filt<sub>T,CSC</sub></b>	Filter parameters for T <sub>CSC</sub>		
			<b>Filt<sub>T,EU</sub></b>	Filter parameters for T <sub>EU</sub>		
			<b>s<sub>0</sub></b>	Sensitivity vector, at 0 °C and t = t <sub>0</sub>	eu/nT	±2
			<b>s<sub>t</sub></b>	Sensitivity vector change with time	eu/ (nT yr)	±10 <sup>-3</sup>
			<b>s<sub>CSC</sub></b>	Sensitivity vector change with sensor sphere temperature	eu/ (nT °C)	±10 <sup>-4</sup>
			<b>s<sub>CDC</sub></b>	Sensitivity vector change with detector temperature	eu/ (nT °C)	±10 <sup>-4</sup>
			<b>s<sub>EU</sub></b>	Sensitivity vector change with EU temperature	eu/ (nT °C)	±10 <sup>-4</sup>
			<b>t<sub>0</sub></b>	Base time for time changes	UTC	
			<b>t<sub>d</sub></b>	Timestamp, instrument processing and filter group delays for each component	s	±10
			<b>t<sub>T,CSC</sub></b>	Latency of T <sub>CSC</sub> measurements	s	±10 <sup>3</sup>
			<b>t<sub>T,CDC</sub></b>	Latency of T <sub>CDC</sub> measurements	s	±10 <sup>3</sup>
			<b>t<sub>T,CDC_s</sub></b>	Latency of T <sub>CDC</sub> for scaling (s <sub>CDC</sub> )	s	±10 <sup>3</sup>
			<b>t<sub>T,EU</sub></b>	Latency of T <sub>EU</sub> measurements	s	±10 <sup>3</sup>
			<b>u<sub>1</sub>, u<sub>2</sub>, u<sub>3</sub></b>	Sensor non-orthogonality angles	"	±10 <sup>3</sup>
			<b>k<sub>2</sub></b>	2 <sup>nd</sup> order linearization factors	1/eu	±10 <sup>-4</sup>
			<b>k<sub>3</sub></b>	3 <sup>rd</sup> order linearization factors	1/eu <sup>2</sup>	±10 <sup>-9</sup>
			L 1 B P ·		<b>Dist_Max</b>	Total maximum magnetic disturbance vector at VFM sensor
EstW eights	<i>Specification of apriori weights for TCF.VFM parameter estimation</i>					
	Min_ Samples	Minimum number of samples for doing TVF.VFM estimation				100 – 10 <sup>5</sup>

V F M	Bias	Bias weighth		0 – 10 <sup>6</sup>
	Scale	Scale weighth		0 – 10 <sup>6</sup>
	n_orth	Non-orthogonality weighth		0 – 10 <sup>6</sup>
	A	Component variance amplification		0 – 100
	Limit	Limit of component variance effect		0.1 – 100
	Huber_factor	Huber weight factor		1 – 100
	Recalc_Limit	Limit for parameter change triggering re-calculation		0 – 10 <sup>4</sup>
	Limits	Structure holding limits for outlier detection and/or rejection		
	Max_ASM_VFM_Diff	Maximum ASM – VFM scalar discrepancy, for flagging	nT	0 – 100
	Reduce_Pars	Parameters used in the sampling reduction algorithm		
TCF_Aux	Structure holding auxiliary TCF.VFM records, see [AD03]			
AOCS		Structure characterizing the magnetic stray field vectors of the AOCS torquer coils at the VFM sensor and various other positions. Includes non-linearities, AOCS-coil and VFM transfer functions, and maximum disturbance vector, see Table 6-3 in Section 6.1.2		
ASM.Filter	Coeff	ASM instrument filter coefficients		±5
	Delay	ASM filter group delay	s	0 – 5
Bus		Structure characterizing the magnetic field of S/C platform: bus currents, induced, states, and static fields at VFM sensor (and other) positions. See Table 6-3 in Section 6.1		
STR.VFM		Stray magnetic field of STR at VFM sensor position	nT	±10
Structure. STR_q_VFM		Rotation of CRF frame to VFM sensor i.e. VFM→CRF transformation cf. Table 3-1		
Structure. STR_q_VFM_Err		Error estimate of STR_q_VFM	"	0 – 60

	Structure. <b>VFM_q</b>	Rotation of VFM sensor frame to S/C frame		
TCF.VFM · M i n P u t	<i>TCF.VFM record from previous Level 1b Product</i>			
	<b>bias</b>	Temporal biases	nT	±100
	<b>scale</b>	Temporal scales		±2
	<b>n_orth</b>	Temporal non-orthogonality angles	"	±50
	<b>rms</b>	Rms misfit (after estimation)	nT	0 – 10
	<b>cov</b>	Parameter covariance matrix (9x9)		0 – 1
	<b>samples</b>	Number of samples used in estimation		0 – 10 <sup>5</sup>
	<b>trange</b>	Time range of data used in estimation	UTC	
	<b>W_scale</b>	Scaling weights used		
	DPU_id	VFM unit identifier		2, 3

<b>Outputs</b>				
<b>Element</b>		<b>Description</b>	<b>Unit</b>	<b>Range</b>
Level 1b <sub>Inst</sub> · <b>VFM</b>		VFM measurements converted using CCDB.VFM.Cal parameters, not corrected for stray fields	nT	± 66,000
<b>B'</b> <sub>VFM</sub>		Fully converted and corrected magnetic field, computed using TCF.VFM <sup>init</sup> parameters	nT	± 66,000
<b>t</b> <sub>out,VFM</sub>		Output time-instants of VFM measurements, 50 Hz	UTC	
TCF.VFM <sub>output</sub>		Structure as on input with updated fields		
Level 1b. Mag-H (50 Hz)	<b>B</b> <sub>VFM</sub>	Fully converted and corrected magnetic field vectors in orthogonal VFM sensor frame at <b>t</b> <sub>out,VFM</sub> time instants	nT	± 66,000
	<b>B</b> <sub>NEC</sub>	Fully converted and corrected magnetic field vector in NEC <sub>VFM</sub> frame, <b>t</b> <sub>out,VFM</sub> time instants	nT	± 66,000
	<b>B</b> <sub>error</sub>	Vector error estimate of <b>B</b> <sub>VFM</sub>	nT	0 – 100
	<b>dB</b> <sub>Sun</sub>	Sun induced stray field correction	nT	±10
	<b>dB</b> <sub>AOCS</sub>	AOCS stray field correction	nT	±50
	<b>dB</b> <sub>other</sub>	Stray field correction – all other sources	nT	±20
	<b>Flags</b> <sub>B</sub>	Flags related to the VFM measurements		
<b>Flags</b> <sub>Platform</sub>	Flags related to the S/C platform			

Level 1b. Mag-L (1 Hz)	$\mathbf{B}_{\text{VFM}}$	Fully converted and corrected magnetic field vectors in orthogonal VFM sensor frame at UTC seconds	nT	$\pm 66,000$
	$\mathbf{B}_{\text{NEC}}$	Fully converted and corrected magnetic field vector in $\text{NEC}_{\text{VFM}}$ frame, UTC seconds	nT	$\pm 66,000$
	$\mathbf{B}_{\text{error}}$	Vector error estimate of $\mathbf{B}_{\text{VFM}}$	nT	0 – 100
	$d\mathbf{B}_{\text{Sun}}$	Sun induced stray field correction	nT	$\pm 10$
	$d\mathbf{B}_{\text{AOCS}}$	AOCS stray field correction	nT	$\pm 50$
	$d\mathbf{B}_{\text{other}}$	Stray field correction – all other sources	nT	$\pm 20$
	Flags <sub>SB</sub>	Flags related to the VFM measurements		
	Flags <sub>Platform</sub>	Flags related to the S/C platform		
Level 1b.Mag-C		Level 1b calibration and verification data		
VFMXAUX_1B (Level 1b)		Auxiliary Level 1b product containing magnetic stray fields at VFM sensor position		

### 6.3.3 Algorithm Overview

The overall VFM processing is sketched in Figure 6-7 below. Only data with a common DPU\_id is to be processed, minority data (DPU\_id wise) are rejected. Data with  $\text{VFM}_{\text{time}}$  less than 20,000,000 (data from first ~2.5 second after VFM switch-on) are also rejected. Next, the raw output (Level 1a.VFM.Sci.EU, rate: 50 Hz, timestamps  $t_{0,\text{VFM}}$ ) of the VFM is corrected for time-stamp, processing, and filter delays. Then, it is corrected and converted to physical units (nT) using the CCDB.VFM.Cal parameters and the Level 1a.VFM.Sci.T<sub>x</sub> temperatures. This is the Level 1b<sub>inst</sub> VFM vector product. The rate is 50 Hz and the time-instants  $t_{\text{VFM}}$ .

Then, the preliminary vector field measurements,  $\mathbf{B}'_{\text{VFM}}$ , are computed using the TCF.VFM<sup>init</sup> parameters. TCF.VFM<sup>init</sup> is the most recent (w.r.t. data being processed) TCF.VFM record among the TCF.VFM<sup>input</sup> (from the previous Level 1b Product) and the CCDB.L1BP.VFM.TCF\_Aux array – with DPU\_id corresponding to Level 1a.VFM.Sci.DPU\_id. Formally, let  $t_0 = \text{Level 1a.VFM.Sci.t}_0$ , then:

$$\text{TCF.VFM}^{\text{init}} = \begin{cases} \text{TCF.VFM}^{\text{input}} & \text{unless ...} \\ \text{TCF\_Aux}_i & \text{if } \text{TCF.VFM}^{\text{input}}.\text{DPU\_id} \neq \text{Level 1a.VFM.Sci.DPU\_id} \\ & \text{or } t_0 \geq \text{TCF\_Aux}_i.\text{trange}_{\text{min}} > \text{TCF.VFM}^{\text{input}}.\text{trange}_{\text{min}} \\ & \text{with } \text{TCF\_Aux}_i.\text{DPU\_id} = \text{Level 1a.VFM.Sci.DPU\_id}; \\ & \text{TCF\_Aux}_i \text{ is the latest TCF\_Aux record (trange}_{\text{min}} \text{ wise)} \\ & \text{for which the above holds.} \end{cases} \quad (6.3-5)$$

Next, outliers are detected and accordingly rejected or flagged as suspicious samples.

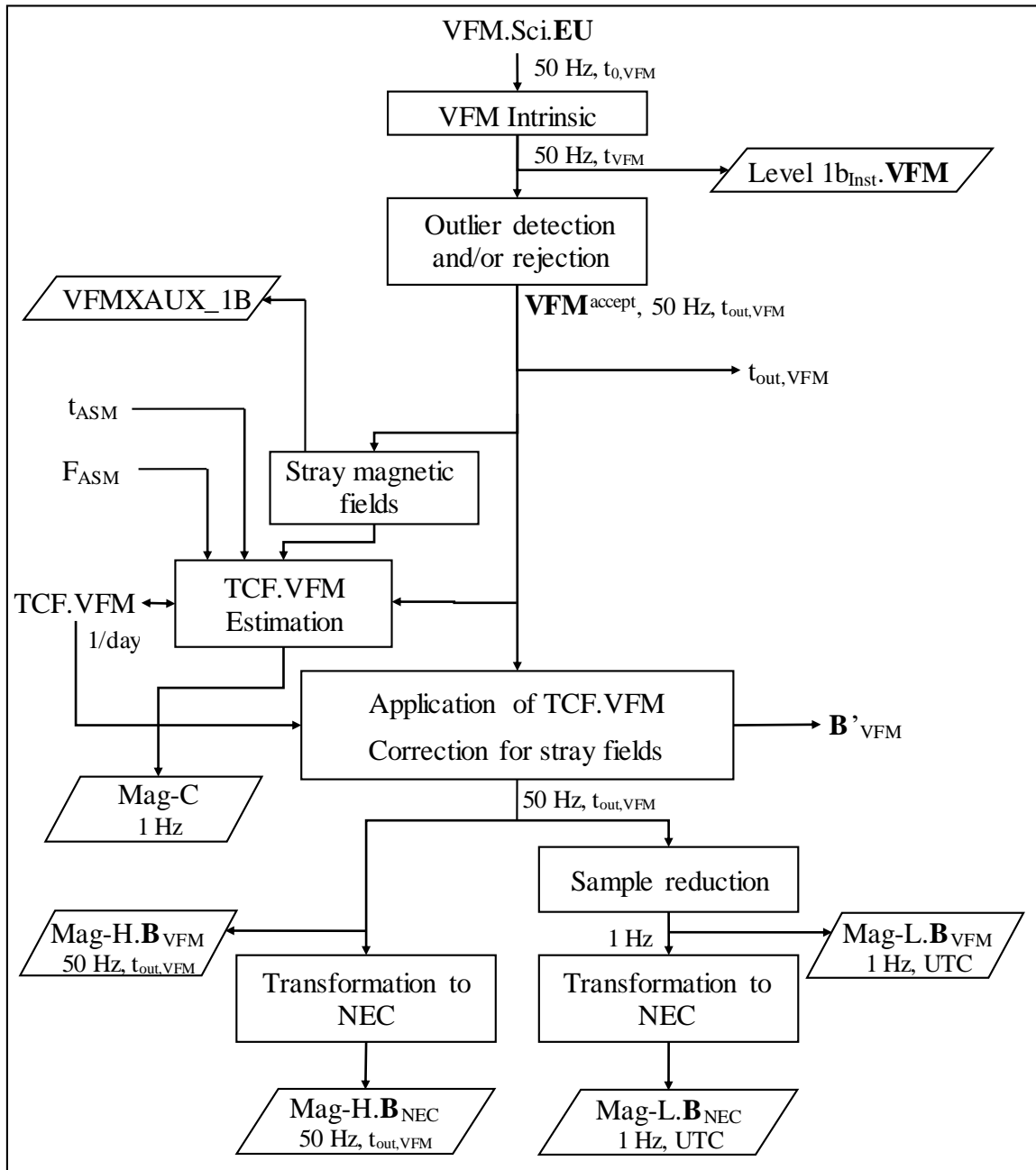


Figure 6-7 VFM Processing Overview

Now, the magnetic stray fields from the rest of the S/C at the VFM sensor position – at the time instants of the VFM measurements,  $t_{out,VFM} (= t_{VFM})$  – are computed. The phase linearity and fast response of the VFM Bessel filter makes it unnecessary to apply a filter to the stray fields as was the case for the ASM. However, the characterization of the AOCS torquer coil disturbances possibly needs to take the VFM filter into account.

Next, the internal temporal changes of the VFM electronics<sup>8</sup> and possibly any change in the non-orthogonalities of the CSC are modeled by the TCF.VFM parameters which are

<sup>8</sup> Due to radiation and age, not modeled by the CCDB.VFM.Cal parameters.



estimated by comparison of VFM data with the fully corrected ASM scalar data ( $F_{ASM}$ ).

The new estimates of the TCF.VFM parameters, TCF.VFM<sup>output</sup>, are applied to all the VFM data of the actual day. Together with the correction for stray magnetic fields the Level 1b.Mag-H. $\mathbf{B}_{VFM}$  data are obtained. I.e. fully converted and corrected magnetic vector data in the orthogonal VFM sensor frame. The rate is 50 Hz and the time instants  $t_{out,VFM}$ .

Then Level 1b.Mag-L. $\mathbf{B}_{VFM}$ , the 1 Hz magnetic vector product in the VFM frame at UTC seconds, is extracted.

Finally, the magnetic field vectors  $\mathbf{B}_{VFM}$  at 50 Hz and 1 Hz are transformed via the Common Reference Frame (CRF) into the NEC frame. This completes the generation of the Level 1b.Mag-H. $\mathbf{B}_{NEC}$  and Level 1b.Mag-L. $\mathbf{B}_{NEC}$  products.

Note: the rotation from the VFM to the CRF frame is estimated pre-flight and refined in-flight. Hence periodic updates of CCDB.Structure.STR\_q\_VFM are foreseen and consequently reprocessing of the final step above is required regularly. See also Section 4.2.

### 6.3.4 Detailed Description of VFM Processing

#### 6.3.4.1 Intrinsic VFM Processing

First, the timestamps are corrected for any processing and filter delays:

$$t_{VFM} = t_{0,VFM} - \text{median}(\text{CCDB.VFM.Cal.t}_d) \quad (6.3-10)$$

Then the instrument calibration parameters of CCDB.VFM are applied to convert the measurements into physical units (nT) and into an orthogonal frame defined as the VFM frame. This follows closely the description in [AD06]. Initially, the linearization terms are applied and the components are linearly interpolated to the unified time-instants,  $t_{VFM}$  (in the following, the CCDB.VFM.Cal structure prefix has been omitted for clarity – all parameters are elements of this structure):

$$E_i = \text{Interpolate}(t_{0,VFM} - t_{d,i}, 1 + ((k_{2,i} + k_{3,i} \text{EU}_i) \text{EU}_i) \text{EU}_i, t_{VFM}) \quad (6.3-20)$$

where  $\text{EU}_i$  are the components of Level 1a.VFM.Sci.EU.

The rest of the conversion is conveniently described in matrix form:

$$\text{Level 1b}_{inst.VFM} = \mathbf{P}^{-1} \times \mathbf{S}^{-1} \times (\mathbf{E} - \mathbf{b}) \quad (6.3-30)$$

The bias vector,  $\mathbf{b}$ , is given by:

$$\mathbf{b} = \mathbf{b}_0 + \frac{\mathbf{b}_t(t - t_0)}{365 \cdot 86400 \text{ s/yr}} + \mathbf{b}_{EU} T_{EU} + \mathbf{b}_{CDC} T_{CDC} + \text{Interpol}_{Lin}(\text{deltab}_{CDC}, T_{CDC}) \quad (6.3-40)$$

where  $t$  is the time of the measurement to convert,  $T_{XXX}$  are de-multiplexed, filtered, time-shifted, linearly interpolated values of Level 1a.VFM.Sci.T<sub>X</sub>, i.e.  $T_{XXX}$  are values of linear interpolation in time to the  $t_{VFM}$  time-instants of  $T'_{XXX}$ :

$$T_{XXX} = \text{Interpol}_{Lin}(T'_{XXX}, t_{VFM} + \text{CCDB.VFM.Cal.t}_{T,XXX}) \quad (6.3-43)$$

where

$$T'_{XXX} = \text{Filter}(\text{CCDB.L1BP.VFM.Cal.Filt}_{XXX}, \text{de-multi}(\text{Level 1a.VFM.Sci.T}_{XXX})) \quad (6.3-45)$$

The  $\text{Interpol}_{Lin}$  and  $\text{Filter}$  functions of Eq. (6.3-40) and (6.3-45) are defined in Appendix E.

The diagonal scaling matrix,  $S^{-1}$ , is:

$$S^{-1} = \begin{pmatrix} s_1 & 0 & 0 \\ 0 & s_2 & 0 \\ 0 & 0 & s_3 \end{pmatrix}^{-1} = \begin{pmatrix} 1/s_1 & 0 & 0 \\ 0 & 1/s_2 & 0 \\ 0 & 0 & 1/s_3 \end{pmatrix} \quad (6.3-50)$$

where  $\mathbf{s} = (s_1, s_2, s_3)$  is given by:

$$\mathbf{s} = \mathbf{s}_0 + \frac{\mathbf{s}_t(t - t_0)}{365 \cdot 86400 \text{ s/yr}} + \mathbf{s}_{CSC} T_{CSC} + \mathbf{s}_{CDC} T_{CDC,s} + \mathbf{s}_{EU} T_{EU} \quad (6.3-60)$$

$T_{CSC}$  and  $T_{CDC,s}$  are computed according to Eq. (6.3-43) (with  $X_{xx} = CSC$  and  $CDC,s$  respectively).

Finally, the matrix converting to the orthogonal VFM frame,  $P^{-1}$ , is:

$$P^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ -\sin u_1 & \cos u_1 & 0 \\ \sin u_2 & \sin u_3 & \sqrt{1 - \sin^2 u_2 - \sin^2 u_3} \end{pmatrix}^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ \sin u_1 / \cos u_1 & 1 / \cos u_1 & 0 \\ -\frac{\sin u_1 \sin u_3 + \cos u_1 \sin u_2}{w \cos u_1} & -\frac{\sin u_3}{w \cos u_1} & \frac{1}{w} \end{pmatrix} \quad (6.3-70)$$

where  $w = \sqrt{1 - \sin^2 u_2 - \sin^2 u_3}$ . Note:  $u_i$  is given in arc-seconds (") and shall be converted, e.g.  $\sin u_1 = \sin_{\text{rad}}(u_1 / 3600 * \pi/180)$ .

Define the uncertainty:  $\mathbf{VFM}_{\text{err}} = (0.03, 0.03, 0.03) \text{ nT}$ .

### 6.3.4.2 Outlier Detection

The VFM outlier detection is similar to the ASM procedure. First, the Level 1b<sub>Inst.</sub>  $\mathbf{VFM}$  ( $\mathbf{VFM}$  for short) are converted to scalar values,  $F^{\mathbf{VFM}} = |\mathbf{VFM}|$  upon which a simple range check is performed. Then, the acceleration of both vector components as well as of the scalar response are checked.

The CCDB.L1BP.VFM.Limits structure contains the following fields:

Field	Type	Unit	Comment
Range	2-d vector	nT	Allowed range, e.g. [14000, 66000]
Max_Accel	real	nT/s <sup>2</sup>	Maximum allowed acceleration of vector components
Max_F_Accel	real	nT/s <sup>2</sup>	Maximum allowed acceleration of scalar response
Pass_Accel	real	nT/s <sup>2</sup>	Vector component accelerations below this are accepted Between Pass_Accel and Max_Accel samples are flagged and associated with raised uncertainty estimates

Field	Type	Unit	Comment
Pass_F_Accel	real	nT/s <sup>2</sup>	Scalar field accelerations below this are accepted Between Pass_F_Accel and Max_F_Accel: flag and raise uncertainty estimates
Max_Gap_Size	real	s	Definition of size of data gaps

Table 6-6 CCDB.L1BP.VFM.Limits Structure

First, all  $F^{VFM}$  samples outside Range are discarded. Then, the 1<sup>st</sup> and 2<sup>nd</sup> order time derivatives of the vector components and of the scalar response ( $F^{VFM}$ ) of the remaining samples,  $A'$  respectively  $A''$ , are computed:

$$A'_i = (A_i - A_{i-1}) / (t_i - t_{i-1}) \quad (6.3-80)$$

$$A''_i = 2(A'_{i+1} - A'_i) / (t_{i+1} - t_{i-1}) \quad (6.3-90)$$

Eq. (6.3-80) and (6.3-90) are computed for  $A = VFM_x, VFM_y, VFM_z$ , and  $F^{VFM}$ .

$VFM_{x/y/z}''$  are checked versus Max\_Accel and Pass\_Accel,  $F^{VFM}''_i$  versus Max\_F\_Accel and Pass\_Accel similar as for the ASM outlier detection.

For samples flagged as suspicious, the associated error estimates are raised:

$$VFM_{k,err}^{new} = \sqrt{VFM_{k,err}^2 + \Delta VFM_k^2}, k = 1,2,3 \quad (6.3-100)$$

$$\Delta VFM = VFM_j - (VFM_{j-1} + (VFM_{j+1} - VFM_{j-1}) (t_j - t_{j-1}) / (t_{j+1} - t_{j-1})) \quad (6.3-110)$$

$\Delta VFM$  is the vector difference between  $VFM_j$  and the linearly interpolated value of  $VFM_{j-1}$  and  $VFM_{j+1}$  (using only accepted VFM samples).

Let  $VFM^{accept}$  denote the Level 1b<sub>Inst.</sub> VFM samples that are not rejected.

#### 6.3.4.3 Filter Group Delay Correction

The correction for the VFM instrument filter group delay is included in CCDB.VFM.Cal.t\_d, cf. Eq. (6.3-10), hence for  $VFM^{accept}$  samples:

$$t_{out,VFM} = t_{VFM} \quad (6.3-120)$$

#### 6.3.4.4 Stray Magnetic Fields

The calculation of the magnetic disturbances,  $\Delta B$ , of the VFM measurements is given by Eq. (6.3-130) below. Note: this computation needs to be done at all time instants of  $VFM^{accept}$ ,  $t_{out,VFM}$ , i.e. at 50 Hz. As the filter of the VFM instrument is very fast and very phase linear, this has only to be taken into account for the AOCS torquer coil disturbances. This is handled by using the transfer characteristic of the combined torquer coil and VFM transfer functions stored in the CCDB.AOCS.Transfer<sub>VFM</sub> structure, see Section 6.1.2.

Note: the  $\Delta B_{AOCS,VFM}$  only contains the stray fields of the magneto-torquer coils, it does not include stray fields from latch valves and thrusters – these are contained in  $\Delta B_{State,VFM}$ .

The stray fields from the S/C body and the bus currents are computed as for the ASM (see pp. 52-54 plus Section 6.1). For clarity, the parameters are written without the Level 1a prefix, and **VFM\_q** without the CCDB.Structure prefix:

$$\Delta \mathbf{B} = \text{CCDB}.\text{STR}.\mathbf{VFM} + \text{Interpol}_{\text{Lin}}(\Delta \mathbf{B}_{\text{Sun}}) + \left( \begin{array}{l} \Delta \mathbf{B}_{\text{AOCS,VFM}}(\text{AOCS}.\text{Mag\_HK}.\text{Imea\_MTQ}) + \\ \text{CCDB}.\text{Bus}.\text{Static}.\mathbf{Base}_{\text{VFM}} + \\ \mathbf{M}_{\text{State,VFM}} \times \text{Interpolate}_{\text{Step}}(\text{Bus}.\text{State}.\mathbf{States}) + \\ \Delta \mathbf{B}_{\text{Induced,VFM}} + \\ \text{CCDB}.\text{AOCS}.\text{Thruster}_{\text{VFM}} \times \text{State}_{\text{Thruster}}(\text{AOCS}.\text{Thru\_HK}) + \\ \Delta \mathbf{B}_{\text{Bus,VFM}} \end{array} \right) \quad (6.3-130)$$

where  $\Delta \mathbf{B}_{\text{Sun}}$  is computed for every UTC second (and interpolated linearly in time to  $t_{\text{out,VFM}}$  time instants) as follows: first, the Sun incident angles on the spacecraft, denoted  $\alpha$  and  $\beta$ , are computed as described below where  $\mathbf{r}$  is the S/C position and  $\mathbf{r}_{\text{Sun}} = \mathbf{x}_{\text{l\_sun}}(t)$  the Sun position (both in ITRF).  $\Delta \mathbf{B}_{\text{Sun}}$  is then computed as a 2-D linearly interpolation in  $\text{CCDB}.\text{L1BP}.\text{VFM}.\text{dB\_Sun}$ , cf. Eq. (6.3-134).

$$\mathbf{q}_{\text{S/C} \leftarrow \text{ITRF}} = \mathbf{q}_{\text{ITRF} \leftarrow \text{ICRF}}^{-1} \times \mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}} \times \text{CCDB}.\text{Structure}.\mathbf{STR\_q}^{-1} \quad (6.3-131)$$

$$\mathbf{s} = \mathbf{q}_{\text{S/C} \leftarrow \text{ITRF}} \times (\mathbf{r}_{\text{Sun}} - \mathbf{r}) \quad (6.3-132)$$

$$\alpha = \text{atan2}(s_z, s_x), \quad \beta = \text{atan2}(-s_y, \sqrt{s_x^2 + s_z^2}) \quad (6.3-133)$$

$$\Delta \mathbf{B}_{\text{Sun,VFM}}^{\text{look-up}} = \text{Interpol}_{\text{Lin}}(\alpha, \beta, \text{CCDB}.\text{L1BP}.\text{VFM}.\text{dB\_Sun}) \quad (6.3-134)$$

$$\Delta \mathbf{B}_{\text{Sun,[VFM]}} = \sum_{i=1}^n (b_i \Delta \mathbf{B}_{\text{Sun,VFM}}^{\text{look-up}}(t - i + 1)) - \sum_{i=2}^n (a_i \Delta \mathbf{B}_{\text{Sun,VFM}}(t - i + 1)) \quad (6.3-134a)$$

where  $a$  and  $b$  are taken from  $\text{CCDB}.\text{L1BP}.\text{VFM}.\text{dB\_Sun\_Filter}$ .

Gaps in positions and/or quaternions are filled with cubic inter-/extrapolation, up to  $\pm 1$  minute for quaternions and  $\pm 10$  minutes for position; outside these limits,  $\alpha$ ,  $\beta$ , and  $\Delta \mathbf{B}_{\text{Sun}}$  are all set to zero.

$\Delta \mathbf{B}_{\text{Bus,VFM}}$  (in Eq. (6.3-130)) is computed as described in Section 6.1.2 using filtered, linearly interpolated Level 1a.Bus.Curr. $\mathbf{I}_{\text{group}}$  currents (at  $t_{\text{out,VFM}}$  time instants), and  $\Delta \mathbf{B}_{\text{Induced,VFM}}$  is computed as:

$$\Delta \mathbf{B}_{\text{Induced,VFM}} = (\text{CCDB}.\text{Bus}.\text{Static}.\text{Ind}_{\text{VFM}} + \sum_i \text{Level1a}.\text{Bus}.\text{State}.\text{IndState}_i \cdot \text{CCDB}.\text{Bus}.\text{State}.\text{Induced}_i.\text{dInd}_{\text{VFM}}) \times (\mathbf{VFM\_q}^{-1} \times \mathbf{B}'_{\text{VFM}} + \Delta \mathbf{B}_{\text{Bus,Body}}) \quad (6.3-135)$$

$\mathbf{B}'_{\text{VFM}}$  is calculated by Eq. (6.3-180) below and  $\Delta \mathbf{B}_{\text{Bus,Body}}$  as described in Section 6.1.2. The recursion in the calculation of  $\Delta \mathbf{B}$  is avoided by setting  $\Delta \mathbf{B}_{\text{Induced,VFM}} = 0$  in the  $\mathbf{B}'_{\text{VFM}}$  term in Eq. (6.3-135).

$\Delta \mathbf{B}$  is used in the following subsections. The various elements of  $\Delta \mathbf{B}$  are transformed to the VFM sensor frame by  $\text{CCDB}.\text{Structure}.\mathbf{VFM\_q}$  (except  $\text{CCDB}.\text{STR}.\mathbf{VFM}$  and  $\Delta \mathbf{B}_{\text{Sun}}$ ) and stored in the  $\text{VFMXAX}_1\text{B}$  Level 1b Product.

The uncertainty of  $\Delta\mathbf{B}$ ,  $\Delta\mathbf{B}_{\text{err}}$ , reflects any missing telemetry required to compute  $\Delta\mathbf{B}$ . Basically,  $\Delta\mathbf{B}_{\text{err}}$  grows from  $\mathbf{0}$  (where all telemetry is available) – piecewise – linearly up to the maximum  $\Delta\mathbf{B}$  possible. Specifically,  $\Delta\mathbf{B}_{\text{err}}$  is computed for all  $t_{\text{out,VFM}}$  time-instants as:

$$\Delta\mathbf{B}_{\text{err}} = \alpha_{\text{Sun}} \Delta\mathbf{B}_{\text{Sun}}^{\text{Max}} + \alpha_{\text{AOCS}} \Delta\mathbf{B}_{\text{AOCS}}^{\text{Max}} + \alpha_{\text{Curr}} \Delta\mathbf{B}_{\text{Bus.Curr}}^{\text{Max}} + \alpha_{\text{State}} \Delta\mathbf{B}_{\text{Bus.State}}^{\text{Max}} \quad (6.3-140)$$

where  $\Delta\mathbf{B}_{\text{XXX}}^{\text{Max}} = \text{CCDB.XXX.Max}_{\text{VFM}}$  or  $\text{CCDB.L1BP.VFM.dB\_Sun\_Max}$  for  $\text{XXX} = \text{Sun}$ . The formulas for  $\alpha_{\text{XXX}}$  are given next.

Given a time  $t \in t_{\text{out,VFM}}$ , let  $dt_{\text{Pos}}$  and  $dt_{\text{Att}}$  be the respective differences in time between  $t$  and the time-instants of the nearest position respectively attitude information, then

$$\alpha_{\text{Sun}} = \begin{cases} 0 & \text{if } |dt_{\text{Pos}}| < 60s \\ & \text{and } |dt_{\text{Att}}| < 10s \\ \min\left(1, \frac{\max(0, |dt_{\text{Pos}}| - 60s) / 600s + \max(0, |dt_{\text{Att}}| - 10s) / 60s}{\max(0, |dt_{\text{Pos}}| - 60s) / 600s + \max(0, |dt_{\text{Att}}| - 10s) / 60s}\right) & \text{otherwise} \end{cases} \quad (6.3-145)$$

Let  $t_i \leq t \leq t_{i+1}$ , where  $t_i$  are the time-instants of Level 1a.AOCS.  $\text{Mag\_HK.Imea\_MTQ}$  telemetry, then

$$\alpha_{\text{AOCS}} = \begin{cases} 0 & \text{if } t - t_i \leq 1s \\ 1 & \text{otherwise} \end{cases} \quad (6.3-150)$$

Let  $t_i \leq t \leq t_{i+1}$ , where  $t_i$  are the time-instants of Level 1a.Bus.Curr, then

$$\alpha_{\text{Curr}} = \begin{cases} 0 & \text{if } t - t_i \leq 3s \text{ or } t_{i+1} - t \leq 3s \\ \frac{t - t_i - 3s}{6s} & \text{if } t_i + 3 < t < \min(t_i + 9s, t_{i+1} - 9s) \\ \frac{t_{i+1} - t - 3s}{6s} & \text{if } \max(t_i + 9s, t_{i+1} - 9s) < t < t_{i+1} - 3s \\ 1 & \text{if } t_i + 9s \leq t \leq t_{i+1} - 9s \end{cases} \quad (6.3-170)$$

Let  $t_i \leq t \leq t_{i+1}$ , where  $t_i$  are the time-instants of Level 1a.Bus.State, then

$$\alpha_{\text{State}} = \begin{cases} 0 & \text{if } t - t_i \leq 4s \\ 1 & \text{otherwise} \end{cases} \quad (6.3-175)$$

#### 6.3.4.5 Computation of $\mathbf{B}'_{\text{VFM}}$

The computation of the fully converted – using the  $\text{TCF.VFM}^{\text{init}}$  parameters – and stray field corrected magnetic field vector measurements,  $\mathbf{B}'_{\text{VFM}}$ , is:

$$\mathbf{B}'_{\text{VFM}} = \mathbf{P}^{-1} \times \mathbf{S}^{-1} \times (\mathbf{VFM}^{\text{accept}} - \mathbf{b}') - \Delta\mathbf{B} \quad (6.3-180)$$

where  $\mathbf{b}' = \text{TCF.VFM}^{\text{init}}.\mathbf{bias}$  and the  $\mathbf{P}'$  and  $\mathbf{S}'$  matrices are as for the intrinsic VFM processing (Eq. (6.3-50) and (6.3-70)) with  $\mathbf{s}' = \text{TCF.VFM}^{\text{init}}.\mathbf{scale} + \mathbf{s}_\beta \cdot \beta - \mathbf{s}_{\text{exp}}(1 - \exp(-(t-t_0)/\tau)) + \mathbf{s}_t(t-t_0) + \mathbf{s}_{t2}(t-t_0)^2 + \mathbf{ds}_{\text{CSC}} \cdot T_{\text{CSC}} + \mathbf{ds}_{\text{CDC}} \cdot T_{\text{CDC},s} + \mathbf{ds}_{\text{EU}} \cdot T_{\text{EU}}$  and  $(u'_1, u'_2, u'_3) = \text{TCF.VFM}^{\text{init}}.\mathbf{n\_orth}$ .<sup>9</sup>  $\mathbf{s}_\beta$ ,  $\mathbf{s}_{\text{exp}}$ ,  $t_0$ ,  $\tau$ ,  $\mathbf{s}_t$ ,  $\mathbf{s}_{t2}$ ,  $\mathbf{ds}_{\text{CSC}}$ ,  $\mathbf{ds}_{\text{CDC}}$  and  $\mathbf{ds}_{\text{EU}}$  are elements of the VFM section of the L1BP part of the CCDB,  $\beta$  is the Sun incident angles on the spacecraft described in Eq. (6.3-133), while  $T_{\text{CSC}}$ ,  $T_{\text{CDC}}$  and  $T_{\text{EU}}$  are the temperatures described in Eq. (6.3-60).

At this point, the ASM processing needs to be performed to obtain the ASM measurements,  $F_{\text{ASM}}$  at  $t_{\text{ASM}}$  time instants.

#### 6.3.4.6 Estimation of TCF.VFM Parameters

The estimation of the TCF.VFM parameters uses the method described in [RD1]. This is an iteratively, reweighted least-squares fit method that minimizes the difference between the ASM-filtered and interpolated scalar values of the corrected VFM measurements,  $|\mathbf{B}_{\text{VFM}}|$ , and the calibrated and corrected ASM measurements,  $F_{\text{ASM}}$ .

First, the set of VFM, ASM measurement pairs, to be used for the estimation process, is selected as depicted in Figure 6-8 below and is as follows: For all ASM samples (time  $t_{\text{ASM}}$ ) select the  $\mathbf{VFM}^{\text{accept}}$  data points less than 0.01 s from  $t_{\text{ASM}}$ . Now, the following conditions are checked:

- There shall be no gaps in the  $\mathbf{VFM}^{\text{accept}}$  data in the preceding 8 seconds and following 30 ms.
- The S/C telemetry required to compute magnetic stray fields shall be available to allow proper stray field computations, i.e.  $\Delta B_{\text{err}} = 0$  for the preceding 8 seconds

Only VFM, ASM pairs for which both conditions are fulfilled are to be used in TCF.VFM estimation process.

When all applicable ASM samples within the current day have been found and the number of samples is greater than  $\text{CCDB.L1BP.VFM.EstWeights.Min\_Samples}$ , the parameters  $\text{TCF.VFM}.\mathbf{bias}$ ,  $\text{TCF.VFM}.\mathbf{scale}$  and  $\text{TCF.VFM}.\mathbf{n\_orth}$  are estimated through an iterative, reweighted least-squares-fit algorithm. The  $\text{TCF.VFM}^{\text{init}}$  parameters are used as a-priori information in the algorithm such that the estimated parameters only have limited deviations from the previous set of parameters – depending on the  $(\text{CCDB.L1BP.VFM.})\text{EstWeights}$  parameters. However, if the a-priori information is based on data much older than the current data, the deviations are allowed to be larger. The weights of the VFM axes bias and scale parameters, in particular the VFM y axis, are also made dependent on the variance of the selected  $\mathbf{VFM}^{\text{accept}}$  unit vector components, as periodical spacecraft maneuvers are planned to enhance the otherwise poor estimation of the y axis parameters. These maneuvers are hence automatically detected by the Level 1b processor through this variance dependent weighting.

<sup>9</sup> The units of  $\tau$ ,  $\mathbf{s}_t$ ,  $\mathbf{s}_{t2}$ , are w.r.t. years and proper conversions of  $(t-t_0)$  will have to be applied as in Eq. (6.3-60).

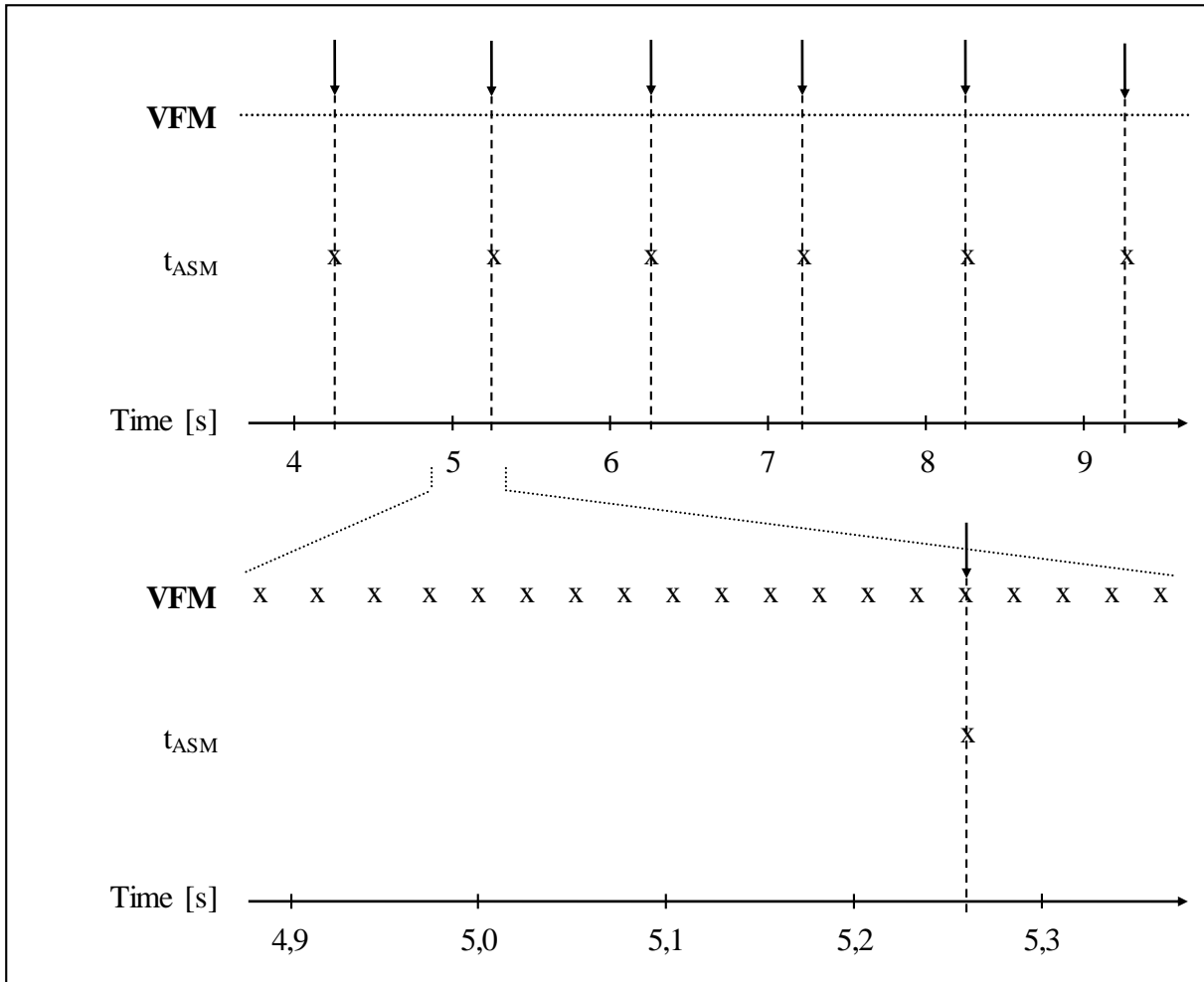


Figure 6-8 Selection of VFM-ASM Data for TCF.VFM Estimation

Basically, the fit minimizes:

$$\chi^2 = \mathbf{d}^T \mathbf{w}^T \mathbf{G} \mathbf{d} + (\mathbf{m} - \mathbf{m}_p)^T \Lambda (\mathbf{m} - \mathbf{m}_p) \quad (6.3-190)$$

where

$\mathbf{m}$  is the model vector of biases, scales and non-orthogonalities (length 9),  $\mathbf{m} = (\mathbf{bias} \ \mathbf{scale} \ as2r \cdot \mathbf{n\_orth})$  with  $as2r = \pi/180/3600$  (arc-seconds to radians).

$\mathbf{m}_p$  is the a-priori model information, i.e.  $\mathbf{m}_p = (\text{TCF.VFM}^{init}.\mathbf{bias} \ \text{TCF.VFM}^{init}.\mathbf{scale} \ as2r \cdot \text{TCF.VFM}^{init}.\mathbf{n\_orth})$ .

$\mathbf{d}$  is the data residual vector (same length as selected  $F_{ASM}$ ):

$$\mathbf{d}_i = F_{ASM,i} - \text{Interpol}_{Lin}(\text{filter}(|\mathbf{B}(\mathbf{m})|), t_{asm,i}) \quad (6.3-200)$$

where filter is the ASM filter applied to the  $|\mathbf{B}(\mathbf{m})|$  50 Hz time-series, linearly interpolated to the  $t_{asm,i}$  time instant. See Eq. (6.2-130) and Eq. (6.2-140) on page 55 how to computer filter.  $\mathbf{B}(\mathbf{m})$  are the converted and corrected VFM samples – at the full, 50 Hz data rate – using model vector  $\mathbf{m}$ :

$$\mathbf{B}(\mathbf{m}) = \mathbf{P}(\mathbf{m})^{-1} \mathbf{S}(\mathbf{m})^{-1} (\mathbf{VFM}^{accept} - \mathbf{b}(\mathbf{m})) - \Delta \mathbf{B} \quad (6.3-210)$$

where

$$\mathbf{b}(\mathbf{m}) = (m_1, m_2, m_3) \quad (6.3-220)$$

with  $w = \sqrt{1 - \sin^2 m_8 - \sin^2 m_9}$  (Note:  $m_7$  through  $m_9$  are in radians.)

and

$$\mathbf{S}^{-1}(\mathbf{m}) = \begin{pmatrix} 1/(m_4 + s_1) & 0 & 0 \\ 0 & 1/(m_5 + s_2) & 0 \\ 0 & 0 & 1/(m_6 + s_3) \end{pmatrix} \quad (6.3-230)$$

with  $\mathbf{s} = (s_1, s_2, s_3)$  given by

$$\mathbf{s} = \mathbf{s}_\beta \beta - \mathbf{s}_{\text{exp}} \left(1 - e^{-(t-t_0)/\tau}\right) + \mathbf{s}_t (t - t_0) + \mathbf{s}_{t2} (t - t_0)^2 + \mathbf{ds}_{\text{CSC}} T_{\text{CSC}} + \mathbf{ds}_{\text{CDC}} T_{\text{CDC},s} + \mathbf{ds}_{\text{EU}} T_{\text{EU}} \quad (6.3-235)$$

where  $\mathbf{s}_\beta$ ,  $\mathbf{s}_{\text{exp}}$ ,  $t_0$ ,  $\tau$ ,  $\mathbf{s}_t$ ,  $\mathbf{s}_{t2}$ ,  $\mathbf{ds}_{\text{CSC}}$ ,  $\mathbf{ds}_{\text{CDC}}$  and  $\mathbf{ds}_{\text{EU}}$  are elements of the VFM section of the L1BP part of the CCDB<sup>10</sup>, and  $\beta$  is the Sun incident angles on the spacecraft described in Eq. (6.3-133), while  $T_{\text{CSC}}$ ,  $T_{\text{CDC}}$  and  $T_{\text{EU}}$  are the temperatures described in Eq. (6.3-60)

and

$$\mathbf{P}^{-1}(\mathbf{m}) = \begin{pmatrix} 1 & 0 & 0 \\ \sin m_7 / \cos m_7 & 1 / \cos m_7 & 0 \\ -\frac{\sin m_7 \sin m_9 + \cos m_7 \sin m_8}{w \cos m_7} & -\frac{\sin m_9}{w \cos m_7} & \frac{1}{w} \end{pmatrix} \quad (6.3-240)$$

$\mathbf{G}$  is the design matrix evaluated at  $\mathbf{m}$ :

$$\mathbf{G} = \frac{\partial \mathbf{d}(\mathbf{m})}{\partial \mathbf{m}} \quad (6.3-250)$$

Since  $\mathbf{d}(\mathbf{m})$  is linear in  $|\mathbf{B}(\mathbf{m})|$  and  $F_{\text{ASM}}$  is constant w.r.t. the partial derivatives:

$$\begin{aligned} G_{ij} &= \text{Interpol}_{\text{Lin}} \left( \text{filter} \left( \frac{-\partial |\mathbf{B}(\mathbf{m})|}{\partial m_j} \right), t_{\text{ASM},i} \right) \\ &\equiv \text{Interpol}_{\text{Lin}} \left( \text{filter}(g_j), t_{\text{ASM},i} \right) \end{aligned} \quad (6.3-260)$$

Since  $|\mathbf{B}(\mathbf{m})| = \sqrt{\mathbf{B}(\mathbf{m})^T \mathbf{B}(\mathbf{m})}$  and

$\mathbf{B}(\mathbf{m})^T = (\mathbf{VFM}^{\text{accept}} - \mathbf{b}(\mathbf{m}))^T \mathbf{S}(\mathbf{m})^{-1} (\mathbf{P}(\mathbf{m})^{-1})^T - \Delta \mathbf{B}^T$ , we get:

$$(g_1, g_2, g_3) = -\mathbf{S}(\mathbf{m})^{-1} (\mathbf{P}(\mathbf{m})^{-1})^T \mathbf{B}(\mathbf{m}) / |\mathbf{B}(\mathbf{m})| \quad (6.3-270)$$

<sup>10</sup> The units of  $\tau$ ,  $\mathbf{s}_t$ ,  $\mathbf{s}_{t2}$ , are w.r.t. years and proper conversions of  $(t-t_0)$  will have to be applied as in Eq. (6.3-60).



$$g_j = \frac{(VFM_{j-3} - m_{j-3})}{(m_j + s_{j-3})} g_{j-3} \quad \text{for } j = 4,5,6 \quad (6.3-280)$$

with  $s$  from Eq. (6.3-235)

$$g_j = \frac{\left( \mathbf{B}(\mathbf{m})^T \frac{\partial \mathbf{P}(\mathbf{m})^{-1}}{\partial m_j} \mathbf{S}(\mathbf{m})^{-1} (VFM_{j-6} - m_{j-6}) + (VFM_{j-6} - m_{j-6})^T \mathbf{S}(\mathbf{m})^{-1} \left( \frac{\partial \mathbf{P}(\mathbf{m})^{-1}}{\partial m_j} \right)^T \mathbf{B}(\mathbf{m}) \right)}{2|\mathbf{B}(\mathbf{m})|} \quad \text{for } j = 7,8,9 \quad (6.3-290)$$

$$\frac{\partial \mathbf{P}(\mathbf{m})^{-1}}{\partial m_7} = \begin{pmatrix} 0 & 0 & 0 \\ 1/\cos^2 m_7 & \sin m_7 / \cos^2 m_7 & 0 \\ -\frac{\sin m_9}{w \cos^2 m_7} & -\frac{\sin m_7 \sin m_9}{w \cos^2 m_7} & 0 \end{pmatrix} \quad (6.3-300)$$

$$\frac{\partial \mathbf{P}(\mathbf{m})^{-1}}{\partial m_8} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -\cos m_8 \frac{\cos m_7 \cos^2 m_9 + \sin m_7 \sin m_8 \sin m_9}{w^3 \cos m_7} & -\frac{\sin m_8 \cos m_8 \sin m_9}{w^3 \cos m_7} & \frac{\sin m_8 \cos m_8}{w^3} \end{pmatrix} \quad (6.3-305)$$

$$\frac{\partial \mathbf{P}(\mathbf{m})^{-1}}{\partial m_9} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -\cos m_9 \frac{\sin m_7 \cos^2 m_8 + \cos m_7 \sin m_8 \sin m_9}{w^3 \cos m_7} & -\frac{\cos^2 m_8 \cos m_9}{w^3 \cos m_7} & \frac{\sin m_9 \cos m_9}{w^3} \end{pmatrix} \quad (6.3-310)$$

$w$  are data weights after Huber (with sigma-factor from CCDB.L1BP.VFM.EstWeights), i.e. initially  $w = \mathbf{1}$  (all ones), for the  $k+1$ 'th iteration given  $\mathbf{d}^k$  and  $w^k$  of the  $k$ 'th iteration:

$$w_i^{k+1} = \min\left(1, \text{EstWeights.Huber\_factor} / |d_i^k / \sigma^k|\right) \quad (6.3-320)$$

with

$$\sigma^k = \sqrt{\sum (w_i^k d_i^k)^2 / \sum w_i^k} \quad (6.3-330)$$

$\Lambda$  is the a-priori model diagonal weight matrix. It is designed to use uniformly scaled CCDB.L1BP.VFM.EstWeights parameters; to adopt a logarithmic decay in the age of the TCF.VFM<sup>init</sup> parameters with a half-time of 5 days, and to adapt to

the variance of the VFM normalized components (to automatically detected S/C maneuvers). Let  $n$  be the number of selected samples of  $F_{ASM}$ ,  $\gamma = (1, 3 \cdot 10^4, 3 \cdot 10^4)$ , and  $t_0$  the time of the first  $\mathbf{VFM}^{accept}$  sample. Then:

$$\Lambda_{ii} = n\lambda \begin{cases} \gamma_1 \eta_i \text{EstWeights.Bias} & \text{for } i=1,2,3 \\ \gamma_2 \eta_{i-3} \text{EstWeights.Scale} & \text{for } i=4,5,6 \\ \gamma_3 \text{EstWeights.n\_orth} & \text{for } i=7,8,9 \end{cases} \quad (6.3-340)$$

$$1/\lambda = 1 + \ln \left( \frac{|\text{TCF.VFM}^{init} \cdot \text{trange}_2 - t_0| + c}{c} \right) \quad (6.3-350)$$

$$c = \frac{5 \cdot 86400 \text{ s}}{e - 1} \quad (6.3-360)$$

The component variance weight factor,  $\eta$ , is computed for all three components:

$$\eta_i = \min \left( \text{EstWeights.Limit}_i, \frac{2n \text{EstWeights.A}}{\sum \left( \frac{\text{VFM}_i}{|\mathbf{VFM}|} \right)^2} \right) \quad (6.3-365)$$

where the sum is taken over the  $n$  VFM samples selected for the estimation.

Now,  $\mathbf{m}$  is estimated by iteration:  $\mathbf{m}^0 = \mathbf{m}_p$  and

$$\mathbf{m}^{k+1} = \mathbf{m}^k + 0.8 \left( (\mathbf{G}^k)^T \mathbf{w}^k \mathbf{G}^k + \Lambda \right)^{-1} \left( (\mathbf{G}^k)^T \mathbf{w}^k \mathbf{d}(\mathbf{m}^k) + \Lambda (\mathbf{m}_p - \mathbf{m}^k) \right) \quad (6.3-370)$$

where

$$\mathbf{G}^k = \left. \frac{\partial \mathbf{d}(\mathbf{m})}{\partial \mathbf{m}} \right|_{\mathbf{m}=\mathbf{m}^k} \quad (6.3-380)$$

The algorithm is iterated until the weighted (with  $\gamma$ ), change in  $\mathbf{m}$  is infinitesimal (though no more than 30 iterations), i.e. until:

$$\sqrt{\sum \gamma_{[i/3]}^2 (m_i^{k+1} - m_i^k)^2} < 10^{-4} \quad (6.3-390)$$

This completes the estimation of TCF.VFM. Set the fields of TCF.VFM<sup>output</sup> as follows with  $\mathbf{m}$  and  $\sigma$  the final model vector and weighted sigma:

$$\mathbf{bias} = (m_1, m_2, m_3)$$

$$\mathbf{scale} = (m_4, m_5, m_6)$$

$$\mathbf{n\_orth} = (m_7, m_8, m_9) / \text{as2r}$$

$$\text{rms} = \sigma$$

$$\text{cov} = (\mathbf{G}^T \mathbf{G} + \Lambda)^{-1}$$

$$\text{W\_scale} = \log_{10}(\text{diagonal of } \Lambda), \text{ with } \log_{10}(0) \equiv -999$$

**trange** = the range of selected  $t_{ASM}$

samples = the number of selected  $F_{ASM}$

DPU\_id = the DPU\_id of the VFM data

In case the final model-vector,  $\mathbf{m}$ , deviates significantly from the initial vector,  $\mathbf{m}_p = \mathbf{m}^0$ , i.e. if  $\sqrt{\sum \gamma_{i/3}^2 (\mathbf{m}_i - \mathbf{m}_i^0)^2} > CCDB.L1BP.VFM.EstWeights.Recalc\_Limit$ , the processing of ASM and VFM data need to be redone starting at the stray field computation and correction using the  $TCF.VFM^{output}$  as  $TCF.VFM^{init}$ .

### 6.3.4.7 Generation of Mag-C Data

The Mag-C data product is the VFM calibration evaluation, verification and re-estimation data product. It contains the following elements – in case no TCF.VFM estimation is performed, replace  $\mathbf{B}(\mathbf{m})$  by  $\mathbf{B}'_{VFM}$  in the table below: In case there are no ASM data available for the entire day, Mag-C data are generated using  $t_{ASM} = UTC$  seconds for the day, the  $F$ ,  $dF_{AOCS}$ ,  $dF_{other}$ , and  $F_{error}$  values are all set to zero.

Element	Description
t	Time from $t_{ASM}$
$\mathbf{r}$	Position, ITRF, spherical coordinates at time t
F	ASM reference field from $F_{ASM}$ (at time t)
$dF_{AOCS}$	Magneto-torquer stray field correction applied to F
$dF_{other}$	Rest of stray field correction applied to F
$F_{error}$	Error estimate $F_{err}^{(3)}$ , from Figure 6-4
$F_{VFM}$	Fully corrected, ASM-filtered, interpolated VFM field intensity response: $Interpol_{Lin}(filter( \mathbf{B}(\mathbf{m}) ))$ where $\mathbf{m}$ is the final model vector
$dt_{VFM}$	Time offset of the VFM sample nearest to $t_{out,ASM} = t - 3*CCDB.ASM.Filter.Delay$ : $dt_{VFM} = t_{out,VFM} - t_{out,ASM}$
$\mathbf{B}$	$\mathbf{B}(\mathbf{m})$ at time $t_{out,VFM}$ of $dt_{VFM}$ above
$d\mathbf{B}_{Sun}$	$\Delta\mathbf{B}$ at time $t_{out,VFM}$ of $dt_{VFM}$ above, Sun induced part
$d\mathbf{B}_{AOCS}$	$\Delta\mathbf{B}$ at time $t_{out,VFM}$ of $dt_{VFM}$ above, magneto-torquer part
$d\mathbf{B}_{other}$	$\Delta\mathbf{B}$ at time $t_{out,VFM}$ of $dt_{VFM}$ above, non magneto-torquer part
$\mathbf{B}_{pre}$	Pre-calibrated VFM magnetic field vector, $\mathbf{VFM}^{accept}$ , at time $t_{out,VFM}$ of $dt_{VFM}$ above
$\mathbf{EU}_{VFM}$	Level 1a.VFM.Sci.EU used to generate $\mathbf{B}$
$T_{CDC}$	CDC temperature linearly interpolated to time $t_{out,VFM}$ of $dt_{VFM}$ above
$T_{CSC}$	CSC temperature linearly interpolated to time $t_{out,VFM}$ of $dt_{VFM}$ above
$T_{EU}$	EU temperature linearly interpolated to time $t_{out,VFM}$ of $dt_{VFM}$ above

alpha	Solar inclination angle, rotation about $-y$ axis, S/C frame, linearly interpolated to time $t_{out,VFM}$ of $dt_{VFM}$ above
beta	Solar inclination angle, angle to $-y$ axis, S/C frame, linearly interpolated to time $t_{out,VFM}$ of $dt_{VFM}$ above

Table 6-7 Mag-C Product Elements

### 6.3.4.8 Application of TCF and Correction for Stray Fields

The TCF.VFM parameters obtained above are applied to all  $VFM^{accept}$  samples of the current day and the correction for stray magnetic fields is performed yielding the Level 1b.Mag-H. $B_{VFM}$  data. This is done as the computation of  $B'_{VFM}$ , see Eq. (6.3-180)

In case the TCF.VFM parameters are not estimated (due to insufficient number of samples), Level 1b.Mag-H. $B_{VFM} = B'_{VFM}$ , and TCF.VFM<sup>init</sup> rather than TCF.VFM<sup>output</sup> is stored in the TCF.VFM Data Set of the Level 1b Products.

The Level 1b.Mag-H. $B_{error}$  (VFM frame) is computed component-wise as the root-sum-squares of the uncertainties:

$$B_{VFM,err}|_i = \sqrt{\frac{VFM_{err}|_i^2 + (\Delta B_i \text{ CCDB.Bus.Error}_{VFM})^2 + \Delta B_{err}|_i^2 + TCF.VFM^{output \text{ or } init}.rms^2}{2}} \quad (6.3-400)$$

### 6.3.4.9 Sample Reduction

The Level 1b.Mag-H. $B_{VFM}$  vector data are reduced to 1 Hz sample rate at UTC seconds using the following method – observing the introduction of leap seconds, cf. Section 4.6. Elements of the reduction method described in [RD2] are used. The basic procedure is to filter the  $B_{VFM}$  vectors component-wise using the Filter function of Appendix E on Page 148 with the CCDB.L1BP.VFM.Reduce\_Pars.Filter coefficients followed by a fit of Chebychev polynomials to 0.1 – 10 second intervals of data using robustly reweighted iterative least squares. The value of the fitted polynomial at the desired sampling point is reported as the reduced-rate sample. The uncertainty is estimated by a bootstrap variability of the fitted values under repeatedly resampling of the residuals and re-fitting.

After filtering, the detailed processing for each target time  $t$  (UTC second) is:

- Select segment of  $\leq n$  filtered VFM samples,  $n = CCDB.L1BP.VFM.Reduce\_Pars.Window$ , preferably centered at  $t$ , but with some skewness allowed: there shall be at least  $\sqrt{3n}$  samples before and after  $t$ . Up to two or three gaps of 1 – 3 samples are allowed – they are filled with linearly interpolated values to get exactly  $n$  equally spaced samples (drop samples at end).  
Note: some (filtered) VFM samples may be skipped or used twice.
- Fit Chebychev polynomials of degree  $CCDB.L1BP.VFM.Reduce\_Pars.Degree$  to each vector component of the segment using iteratively reweighted least squares with

robust Tukey's biweights. These weights are parameterized by the CCDB.L1BP.VFM.Reduce\_Pars.Weight\_Param parameter.

- Estimate the scale for the weight function using a two-step procedure, starting with the median absolute deviation of the data in a window from their median to get the first scale estimate, then updating the scale estimate once using the median absolute deviation from the first polynomial fitted using weights calibrated to the first scale estimate. In all cases, the scale is only applied if above  $10^{-3}$ . The fit tends to converge in 7-20 iterations, depending in part on details of the weight function.
- Report the value of the fitted polynomial at the target time  $t$  as the sample at the reduced rate, i.e. the Level 1b.Mag-L. $\mathbf{B}_{VFM}$  data point. Estimate the standard error of the fitted polynomial at this point using CCDB.L1BP.VFM.Reduce\_Pars.Bootstraps bootstraps, by pretending that the fitted function is correct and drawing with replacement from the residuals to the fitted function to produce sets of "pseudo-data", fitting those pseudo-data by the same iterative procedure, and reporting the standard deviation of the fitted values.

The error estimate of the 1 Hz product is computed by root-sum-square adding the standard error estimates,  $se$ , to the average, possibly weighted,  $\mathbf{B}_{VFM,err}$  of the  $n$  (real) samples in the segment:

$$\text{Level 1b.Mag - L.}\mathbf{B}_{error}|_i = \sqrt{\left(\frac{\sum \mathbf{B}_{VFM,err}|_i}{n}\right)^2 + se_i^2} \quad (6.3-410)$$

$dB_{AOCs}$  and  $dB_{other}$  of the Level1b.MagL product is filled with linear interpolated values of the corresponding elements of the Level1b.MagH product.

#### 6.3.4.10 Transformation to NEC Frame

The  $\mathbf{B}_{VFM}$  of the Level 1b Mag-H and Mag-L products are transformed to  $\mathbf{B}_{NEC}$ , the  $NEC_{VFM}$  frame of the VFM using  $\mathbf{q}_{NEC \leftarrow CRF}$  and CCDB.Structure.STR\_q\_CRF:

$$\mathbf{B}_{NEC} = \mathbf{q} \times \mathbf{B}_{VFM} \quad (6.3-420)$$

$$\mathbf{q} = \text{CCDB.Structure.STR\_q\_VFM} \otimes \mathbf{q}_{NEC \leftarrow CRF} \quad (6.3-430)$$

The attitude error estimates,  $Att_{error}$ , is the root-sum-square of the input  $q_{error}$  error estimate and CCDB.Structure.STR\_q\_VFM\_err:

$$\text{Level 1b.Mag} \begin{Bmatrix} H \\ L \end{Bmatrix} . Att_{error} = \sqrt{q_{error}^2 + \text{CCDB.STR\_q\_VFM\_err}^2} \quad (6.3-440)$$

### 6.3.4.11 Flags

The flags of the 50 Hz VFM measurements, Level 1b.Mag-H.Flags<sub>B</sub>, are computed as the sum of the values specified in Table 6-8 below for which the associated conditions are true.

Value	Description
1	ASM instrument turned off
2	VFM temperature trouble (any temperature): outlier detected, gap, or not enough samples for filtering
4	Suspicious VFM sample
8	Discrepancy between ASM and VFM measurements, i.e. $  \text{Level 1b.Mag-H.B}_{\text{VFM}}  - \text{Interpolate}_{\text{Cubic}}(\text{F}_{\text{ASM}})  > \text{CCDB.Max\_ASM\_VFM\_Diff}$

Table 6-8 Level 1b Mag-H Flags<sub>B</sub> values

The flags of the 1 Hz VFM measurements, Level 1b.Mag-L.Flags<sub>B</sub>, are slightly different. They are computed as the sum of the values (except 255) specified in Table 6-9 below for which the associated conditions are true. For values 2, 4, and 16 the CCDB.L1BP.VFM.Reduce\_Pars.Window VFM samples used in the sample reduction (previous page) are considered

Value	Description
1	ASM instrument turned off
2	VFM temperature trouble (any temperature): outlier detected, gap, or not enough samples for filtering
4	More than 5 suspicious VFM samples (in sample reduction window)
8	Discrepancy between ASM and VFM measurements: $  \text{Level 1b.Mag-L.B}_{\text{VFM}}  - \text{Level 1b.Mag-L.F}  > \text{CCDB.L1BP.VFM.Max\_ASM\_VFM\_Diff}$
16	Gap in VFM samples (by rejection or missing data)
255	Insufficient VFM data available for generating Level 1b.Mag-L.B <sub>xxx</sub> (not combined with any of the above)

Table 6-9 Level 1b Mag-L Flags<sub>B</sub> values

The computation of Flags<sub>Platform</sub> of either Mag-H or Mag-L are computed as the sum of the values specified in Table 6-10 below for which the associated conditions are true – for Mag-L considering VFM samples in the sampling reduction window as for Flags<sub>B</sub> above.

Value	Description
1	Thruster latch valves open, thrusters not activated
2	Thrusters activated
4	Gap in Bus telemetry, 1 or 2 samples missing <sup>11</sup>
8	Outlier detected in any Bus currents (from Filter function)
16	Not enough data for filtering Bus currents (due to large gap or jump in data)
32	Change in any of the states listed in CCDB.Bus.State.Field and CCDB.Bus.State.Induced according to Bus telemetry, i.e. <b>State</b> <sub>before</sub> ≠ <b>State</b> <sub>after</sub>
64	No Bus telemetry available (for extended period)
128	Gap in AOCS telemetry <sup>11</sup>
256	Position information based on on-board (GPSR) navigational solution

Table 6-10 Level 1b Mag-H/L Flags<sub>Platform</sub> values

---

<sup>11</sup> 1% slack compared to the nominal sampling rate is allowed.

## 6.4 Star Tracker – STR

<sup>12</sup>The Star Tracker is comprised of three Camera Head Units (CHUs) mounted on the inner-most end of the optical bench. Nominally, the attitudes of all three heads are provided simultaneously at 1 Hz rate, however one head is regularly blinded by the Sun leaving the attitudes of just two heads. The attitudes of the 2-3 CHU are combined into one attitude, the attitude of the STR Common Reference Frame (CRF). The combination uses the method described in [RD3]. The attitudes of CRF are then interpolated to obtain the CRF and S/C attitudes at required time instants: 50 Hz VFM measurements, 2 Hz EFI measurements and UTC seconds.

### 6.4.1 Input and Output Parameters

Inputs						
Element			Description	Unit	Range	
L e v e l 1 a	S T R · S c i	Att1	<b>q</b>	Orientation of CHU1 in ICRF frame, i.e. <b>q</b> is the transformation CHU1 ← ICRF		±1
			Info	Structure with additional information, see Table 6-11		
		Att2		Same as Att1 but for CHU2		
		Att3		Same as Att1 but for CHU3		
	STR.HK		$t_{out,CHU}$	Output time-instants of the CHU temperature data	UTC	
			$T_{CHU1}, T_{CHU2}, T_{CHU3}$	Temperatures of CHU1, CHU2 and CHU3	°C	±50
		Bus.OpticalBench	$t_{out,OB}$	Output time-instants of optical bench temperatures	UTC	
	Bus.OpticalBench		$T_{if1}, T_{if2}, T_{if3}$	Temperatures of ObBoomIf	°C	±40
			$T_{rad1}, T_{rad2}, T_{rad3}$	Temperatures of ObSubSys	°C	±40
		AOCS.Prop_HK		Propellant telemetry		
	AOCS.Thru_HK.		AOCS thruster activation			
	$v_{pre}$		Preliminary S/C velocity, ITRF	m/s	±10 <sup>4</sup>	
	$t_{out,EFI}$		Output time-instants of EFI data, 2 Hz	UTC		
	$t_{out,VFM}$		Output time-instants of VFM data, 50 Hz	UTC		

<sup>12</sup> The STR processing description is placed in this Section for historical reasons. The instrument level processing is actually performed in the GPSR and MOD processing context.



<b>Inputs</b>						
<b>Element</b>		<b>Description</b>		<b>Unit</b>	<b>Range</b>	
	Level 1b.Eph.r	S/C position, ITRF		m	$\pm 10^7$	
C C D R B	S T R	AberrationalCor	Application of aberrational correction		0 – 4	
		Delay	Delay of time-stamp	s	$\pm 0.2$	
		Kappa	Accuracy of rotation about boresight expressed in units of pointing accuracy		1 – 10	
	L 1 B P : S T R	MinConform	Level of conformity required between pairs of CHU attitudes after rotation to CRF with <b>STR_q_CRF</b>			0.9 – 1
		Spline_Pars	B-spline parameters for CRF interpolation			
		CHU1	Modelling parameters for CHU1 (NB. First row has unit [“])		“/°C	$10^4 - 10$
		CHU2	Same as CHU1 but for CHU2		“/°C	$10^4 - 10$
		CHU3	Same as CHU1 but for CHU3		“/°C	$10^4 - 10$
		Filter_T	Filter parameters for CHU temperatures			
	S t r u c t u r e	<b>STR_q</b>		Rotation of CRF to S/C frame		$\pm 1$
STR_q_CHU		Mean rotation of CRF to CHUs frame		$\pm 1$		
Opt ical Be nch u r e		wT_if	Weights for temperature averaging			
		wT_rad	Weights for temperature averaging			
		Filter_if	Filter parameters for OB temperatures			
		Filter_rad	Filter parameters for OB temperatures			
SC_CoG		Structure with parameters for calculating the position of CoG in S/C coordinates as a function of propellant levels				
<b>VFM_r</b>		Position of VFM sensor, S/C coordinates	m	$\pm 10$		
Constants	c	Speed of light		m/s	$\sim 3 \cdot 10^8$	
	Sun	Structure containing constants for computing Earth velocity about the Sun				

<b>Outputs</b>					
<b>Element</b>		<b>Description</b>		<b>Unit</b>	<b>Range</b>
<b>q<sub>NEC←CRF</sub></b>		Rotation of NEC <sub>VFM</sub> to CRF frame at UTC seconds and at $t_{out,VFM}$ time-instants			$\pm 1$
<b>q<sub>error</sub></b>		Uncertainty estimates of <b>q<sub>NEC←CRF</sub></b>		“	0 – 30

Outputs				
Element		Description	Unit	Range
$\mathbf{q}_{\text{NEC} \leftarrow \text{S/C}}$		Rotation of $\text{NEC}_{\text{VFM}}$ to S/C frame at $t_{\text{out,EFI}}$ time-instants		$\pm 1$
$\dot{\mathbf{q}}_{\text{ITRF} \leftarrow \text{S/C}}$		Time-derivative of S/C $\rightarrow$ ITRF rotation	$\text{s}^{-1}$	$\pm 2$
Level 1b	Eph. $\mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}}$	Rotation of CRF to ICRF frame at STR time instants (Instrument Level 1b data)		$\pm 1$
	Eph. $\mathbf{q}_{\text{ITRF} \leftarrow \text{S/C}}$	Rotation of ITRF to S/C frame at UTC seconds (“S/C Attitude”)		$\pm 1$
	Eph. $q_{\text{error}}$	Uncertainty estimate of Level 1b.Eph. $\mathbf{q}_{\text{ITRF} \leftarrow \text{S/C}}$	”	0 – 30
	Mag-L/H. $\mathbf{r}_{\text{VFM}}$	VFM sensor position, ITRF, 1 and 50 Hz	m	$\pm 10^7$
	Mag-L/H. $\mathbf{q}_{\text{NEC} \leftarrow \text{CRF}}$	STR CRF to $\text{NEC}_{\text{VFM}}$ transformation		$\pm 1$
	Xxx.Flags $_q$	Attitude flags of products Xxx = Eph, Mag-L, Mag-H		

The Level 1a.Sci.Att<sub>i</sub>.Info structure contains the following fields:

Field	Description	Unit	Range
Seq	Sequence: If 1 only fine-tuning of attitude is performed, otherwise initial attitude determination is also performed.		0, 1
Cor	Correction: If 1, aberrational and/or precession/nutation correction is performed.		0, 1
HR	HighRate: If 1, considerable smearing of the image is observed – only half the integration time and half the number of pixels are used		0, 1
BBO	Big Bright Object: If 1, a considerable part of the image is obscured		0, 1
t_ref	Time Ref: If 1, the internal STR time is not synchronized to GPS.		0, 1
Valid	Valid: If 1, the provided attitude is valid.		0, 1
Res	Uncertainty estimate in pointing of Att <sub>i</sub> . $\mathbf{q}$	”	0 – 30

Table 6-11 Info Flags of STR Telemetry

## 6.4.2 Algorithm

### 6.4.2.1 Overview

First, the time-stamps are corrected for any delays:  $t_{\text{out,STR}} = t_{0,\text{STR}} - \text{CCDB.STR.Delay}$ . Then the aberrational correction is verified and if necessary applied. The valid attitudes of the STR are combined into common attitude solutions,  $\mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}}$ , providing the CRF  $\rightarrow$  ICRF

transformations. This is the Level 1b<sub>Inst</sub>.STR product. Then these are interpolated to the time-instants to be used (UTC seconds,  $t_{out,VFM}$ , etc.) using a smoothing, cubic B-spline method. Finally, the various, required output transformations are generated by combining series of transformations.

### 6.4.2.2 Aberrational Correction

The *aberrational* effect is a shift of the observed position of a light source proportional to the relative<sup>13</sup> velocity of the observer (the S/C) perpendicular to the line of sight (of the CHU<sub>i</sub>). The velocity of the S/C in ICRF frame causes *orbital aberration*, whereas the Earth's velocity around the Sun causes *annual aberration*. The S/C velocity is given by  $\mathbf{v}_{pre}$  (and  $\mathbf{r}_{pre}$ ) converted to ICRF:  $\mathbf{v}_{pre}^{ICRF} = \mathbf{Q} \mathbf{R} \mathbf{W} \mathbf{v}_{pre}^{ITRF} + \mathbf{Q} \mathbf{R} d\mathbf{R}/dt \times \mathbf{W} \mathbf{r}_{pre}^{ITRF}$  (see Section 8.4, Eq. 8.4-10 for the matrices Q, R, and W) followed by cubic Lagrange interpolation to  $t_{out,STR}$  time-instants. The Earth velocity,  $\mathbf{v}_{Earth}$  is computed using time,  $t_{out,STR}$ , and the CCDB.Constants.Sun parameters (prefix omitted):

$$\mathbf{v}_{Earth} = \frac{-c1 \sin(a_{ecc})\mathbf{SA} + c2 \cos(a_{ecc})\mathbf{SB}}{1 - \text{Eccentricity} \cos(a_{ecc})} \quad (6.4-95)$$

where

$$a_{ecc} = a_{mean} + \text{Eccentricity} \sin(a_{mean}) \quad (6.4-96)$$

$$a_{mean} = (t_{out,STR} - t_{Perigee}) \text{Mean\_Motion} \quad (6.4-97)$$

The computation of the actual aberrational correction is given in Appendix E.4.

The aberrational correction to apply – the velocity to use in Appendix E.4,  $\mathbf{v}_{obs}$  – is determined by CCDB.STR.AberrationalCor and Att<sub>i</sub>.Info.Cor according to the table below. The “On-ground aberrational correction” flag is only set, when Att<sub>i</sub>.Info.Cor = 0 and AberrationalCor = 1, 2, or 3.

AberrationalCor	$\mathbf{v}_{obs}$		Flag
	Att <sub>i</sub> .Info.Cor = 0	Att <sub>i</sub> .Info.Cor = 1	
0	<b>0</b>	<b>0</b>	0
1	$\mathbf{v}_{Earth} + \mathbf{v}_{pre}^{ICRF}$	<b>0</b>	1 - Att <sub>i</sub> .Info.Cor
2	$\mathbf{v}_{Earth} + \mathbf{v}_{pre}^{ICRF}$	$\mathbf{v}_{pre}^{ICRF}$	1 - Att <sub>i</sub> .Info.Cor
3	$\mathbf{v}_{Earth} + \mathbf{v}_{pre}^{ICRF}$	$\mathbf{v}_{Earth}$	1 - Att <sub>i</sub> .Info.Cor
4	$\mathbf{v}_{Earth} + \mathbf{v}_{pre}^{ICRF}$	$\mathbf{v}_{Earth} + \mathbf{v}_{pre}^{ICRF}$	0

<sup>13</sup> Relative with respect to the speed of light,  $c = 299,792,458$  m/s.

### 6.4.2.3 Temperature Correction

First the three CHU temperatures together with the six Optical Bench temperatures are filtered using the Filter function, see page 148 (Appendix E), ( $X = \text{CHU}_j, \text{if}_j, \text{rad}_j$ , where  $j = 1,2,3$ ):

$$T_{F,X} = \text{Filter}(\text{CCDB}..\text{Filter}_X, \text{Level1a}..T_X) \quad (6.4-98a)$$

The filtered temperatures are linear interpolated to match the timestamps of the STR L1a data:

$$T_X = \text{Interpol}_{\text{Lin}}(t_{F,X}, T_{F,X}, t_{\text{out,STR}}) \quad (6.4-98b)$$

where  $t_{F,X}$  is the output time-instants from the Filter function.

The weighted temperatures  $T_{\text{if}}$  and  $T_{\text{rad}}$  are then calculated using  $\text{CCDB}.\text{Structure}.\text{OpticalBench}.\text{wT}_{\text{if}}$  and  $\text{CCDB}.\text{Structure}.\text{OpticalBench}.\text{wT}_{\text{rad}}$  respectively ( $X = \text{if}$  and  $\text{rad}$ ):

$$T_X = \text{Interpol}_{\text{Lin}}(t_{F,X}, T_{F,X}, t_{\text{out,STR}}) \quad (6.4-98c)$$

Finally the temperature correction is applied to the abarational corrected attitudes  $\text{Att}_i.\mathbf{q}$  where  $i = 1, 2, 3$ :

$$\text{Att}_i.\mathbf{q} \otimes \mathbf{q}_1(\alpha_i) \otimes \mathbf{q}_2(\beta_i) \otimes \mathbf{q}_3(\gamma_i) \quad (6.4-98d)$$

where

$$\begin{aligned} \mathbf{q}_1(\alpha_i) &= (\sin(\alpha_i/2), 0, 0, \cos(\alpha_i/2)) \\ \mathbf{q}_2(\beta_i) &= (0, \sin(\beta_i/2), 0, \cos(\beta_i/2)) \\ \mathbf{q}_3(\gamma_i) &= (0, 0, \sin(\gamma_i/2), \cos(\gamma_i/2)) \end{aligned} \quad (6.4-98e)$$

and  $\alpha_i, \beta_i$  and  $\gamma_i$  are linear combinations of the CHU $_i$  temperature given by:

$$\begin{aligned} \alpha_i &= \sum_{j=1}^4 \text{CCDB}.\text{STR}.\text{TemperatureCor}.\text{CHU}_{i,j,1} \cdot T_j \\ \beta_i &= \sum_{j=1}^4 \text{CCDB}.\text{STR}.\text{TemperatureCor}.\text{CHU}_{i,j,2} \cdot T_j \\ \gamma_i &= \sum_{j=1}^4 \text{CCDB}.\text{STR}.\text{TemperatureCor}.\text{CHU}_{i,j,3} \cdot T_j \end{aligned} \quad (6.4-98f)$$

with  $T = [1, T_{\text{CHU}_i}, T_{\text{if}} - T_{\text{CHU}_i}, T_{\text{rad}} - T_{\text{CHU}_i}]$ .

### 6.4.2.4 Attitude Combination

The combined transformation from ICRF to CRF,  $\text{Level1b}_{\text{Inst}}.\text{STR}.\mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}}$ , is computed using a least squares fit with weights corresponding to the actual uncertainty of the STR attitude data including the anisotropic uncertainty of the estimation of the rotation about the camera boresight axis.

In the following, the  $\text{Level1a}.\text{STR}.\text{Sci}$ ,  $\text{CCDB}.\text{STR}$  and  $\text{CCDB}.\text{Structure}$  prefixes for the data and parameters have been omitted for clarity.

Only valid, consistent and high quality attitudes are used, i.e. the following conditions must be met:

- $\text{Att}_i.\text{Info}.\text{Valid}$  must be 1
- $\text{Att}_i.\text{Info}.\text{HR}$  must be 0

- $\text{Att}_i.\mathbf{q}$  must have unity length, i.e.  $\sum_{j=1}^4 q_j^2 = 1$  – to within given precision, i.e. check

$$\left| \sum_{j=1}^4 q_j^2 - 1 \right| < 2 \cdot 10^{-9}$$

Additionally, for samples with  $\text{Att}_i.\text{Info.BBO} = 1$ , the inter-boresight angle(s) (IBA) between the individual CHU's is checked against the nominal IBA designated by  $\text{CCDB.Structure.STR\_q\_CHU}$ . For optimization purposes, the cosines of the angles are used, since for the inter-boresight angle,  $\text{iba}$ , between two CHU quaternions  $\mathbf{q1}$  and  $\mathbf{q2}$ :

$$\cos(\text{iba}) = (\mathbf{q1}^{-1} \times \mathbf{ez}) \cdot (\mathbf{q2}^{-1} \times \mathbf{ez}) \equiv \text{ib} \quad (6.4-99a)$$

where  $\mathbf{ez} = (0 \ 0 \ 1)^T$  is the unit vector of the z-axis. The nominal “ib” for CHU  $i$  and  $j$ ,  $\text{ib}_{i,j\text{nom}}$ , is

$$\text{ib}_{i,j\text{nom}} = (\text{STR\_q\_CHU}_i \times \mathbf{ez}) \cdot (\text{STR\_q\_CHU}_j \times \mathbf{ez}) \quad (6.4-99b)$$

A CHU sample with  $\text{Att}_i.\text{Info.BBO} = 1$  is rejected if for any  $\text{Att}_j$  where the three bullets above are met (valid, not HR, and unit length) and  $\text{Att}_j.\text{Info.BBO} = 0$ :

$$|\text{ib}_{i,j} - \text{ib}_{i,j\text{nom}}| > \text{CCDB.L1BP.STR.IBdev\_BBO\_Limit} \quad (6.4-99c)$$

otherwise  $\text{Att}_i$  is included.

Additionally, the attitudes from the 3 CHU must be consistent, this is checked below.

Let

$$\tilde{\Lambda}_i = (\mathbf{R}_{\text{CHU}_i \leftarrow \text{CRF}})^T \begin{pmatrix} \tilde{\sigma}_i^2 & 0 & 0 \\ 0 & \tilde{\sigma}_i^2 & 0 \\ 0 & 0 & \kappa^2 \tilde{\sigma}_i^2 \end{pmatrix}^{-1} \mathbf{R}_{\text{CHU}_i \leftarrow \text{CRF}} \quad (6.4-100)$$

where  $\kappa = \text{Kappa}$  and  $\mathbf{R}_{\text{CHU}_i \leftarrow \text{CRF}}$  is the rotation matrix corresponding to the  $\text{STR\_q\_CHU}_i^{-1}$  (the  $i^{\text{th}}$  row of  $\text{CCDB.Structure.STR\_q\_CHU}$ ) parameter and where

$$\tilde{\sigma}_i = \begin{cases} \sigma_i + \text{CCDB.L1BP.STR.sigma}_0, & \text{if } \text{Att}_i.\text{Info.BBO} = 0 \\ \sigma_i + \text{CCDB.L1BP.STR.sigma\_BBO}, & \text{if } \text{Att}_i.\text{Info.BBO} \neq 0 \end{cases} \quad (6.4-105)$$

where  $\sigma_i = \max(\text{Att}_i.\text{Res}, 0.3)$ . Let

$$\mathbf{q}^i = \text{q4positive}(\text{STR\_q\_CHU}_i^{-1} \otimes \text{Att1}.\mathbf{q}^{-1} \otimes \text{Att}_i.\mathbf{q} \otimes \text{STR\_q\_CHU}_i) \quad (6.4-110)$$

where  $\text{q4positive}$  returns the quaternion with positive 4<sup>th</sup> component, i.e:

$$\text{q4positive}(\mathbf{q}) = \begin{cases} \mathbf{q} & \text{iff } q_4 \geq 0 \\ -\mathbf{q} & \text{iff } q_4 < 0 \end{cases} \quad (6.4-115)$$

Then  $\mathbf{q}^i \approx (0 \ 0 \ 0 \ 1)$ , i.e. close to unity, provided the attitudes from CHU1 and CHU $i$  are consistent. If  $|q_4^i| < \text{CCDB.L1BP.STR.MinConform}$  the attitudes are considered inconsistent and only the pair  $(j,k)$  of attitudes with highest  $|q_4^{j,k}| \geq \text{MinConform}$  are used ( $\mathbf{q}^{1,k} \equiv \mathbf{q}^k$ ). If no such pair exist – use the attitude with the lowest  $\sigma_i$ . Let

$$\Delta_i = 2 (q_1^i, q_2^i, q_3^i)^T \quad (6.4-120)$$

then

$$\text{Level 1b}_{\text{Inst}} \cdot \text{STR} \cdot \mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}} = \text{Att1} \cdot \mathbf{q} \otimes \text{STR} \cdot \mathbf{q}_{\text{CHU}_1} \otimes (\boldsymbol{\varepsilon}, \sqrt{1-\boldsymbol{\varepsilon}^2}) \quad (6.4-130)$$

where

$$\boldsymbol{\varepsilon} = \frac{1}{2} (\tilde{\Lambda}_1 + \tilde{\Lambda}_2 + \tilde{\Lambda}_3)^{-1} (\tilde{\Lambda}_2 \Delta_2 + \tilde{\Lambda}_3 \Delta_3) \quad (6.4-140)$$

The description above assumes all three attitudes qualify for use. If Att2 or Att3 is invalid or does not qualify, the products of the disregarded attitude are ignored in Eq. (6.4-140). If Att1 is disregarded,  $\text{Level 1b}_{\text{Inst}} \cdot \text{STR} \cdot \mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}}$  is computed as follows:

$$\mathbf{q}^{2,3} = \text{STR} \cdot \mathbf{q}_{\text{CHU}_2}^{-1} \otimes \text{Att2} \cdot \mathbf{q}^{-1} \otimes \text{Att3} \cdot \mathbf{q} \otimes \text{STR} \cdot \mathbf{q}_{\text{CHU}_3} \quad (6.4-150)$$

$$\Delta_{2,3} = 2 (\mathbf{q}^{2,3}_1, \mathbf{q}^{2,3}_2, \mathbf{q}^{2,3}_3)^T \quad (6.4-160)$$

$$\boldsymbol{\varepsilon}' = \frac{1}{2} (\tilde{\Lambda}_2 + \tilde{\Lambda}_3)^{-1} (\tilde{\Lambda}_3 \Delta_{2,3}) \quad (6.4-170)$$

$$\text{Level 1b}_{\text{Inst}} \cdot \text{STR} \cdot \mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}} = \text{Att2} \cdot \mathbf{q} \otimes \text{STR} \cdot \mathbf{q}_{\text{CHU}_2} \otimes (\boldsymbol{\varepsilon}', \sqrt{1-\boldsymbol{\varepsilon}'^2}) \quad (6.4-180)$$

If only one attitude is applicable,  $\text{Att}_i$ ,  $\text{Level 1b}_{\text{Inst}} \cdot \text{STR} \cdot \mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}} = \text{Att}_i \cdot \mathbf{q} \otimes \text{STR} \cdot \mathbf{q}_{\text{CHU}_i}$ .

This also applies in case none of the attitude pairs complies to the conformity requirement ( $|q_i^4| \geq \text{CCDB.L1BP.STR.MinConform}$ ) in which case the attitude with the lowest  $\sigma$  ( $\text{Att}_i \cdot \text{Res}$ ) and  $\text{BBO} = 0$  is used – if all considered attitudes have  $\text{BBO} = 1$ , the one with the lowest  $\sigma$  is used.

The uncertainty estimate of the combined quaternion is computed as follows:

$$\text{Level 1b}_{\text{Inst}} \cdot \text{STR} \cdot \mathbf{q}_{\text{error}} = \sqrt{3/2} \frac{1}{\sum 1/\sigma_i} \quad (6.4-190)$$

where the sum is taken over the attitudes used in the calculation of  $\text{Level 1b}_{\text{Inst}} \cdot \text{STR} \cdot \mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}}$ . If only one attitude,  $\text{Att}_i$ , is used:

$$\text{Level 1b}_{\text{Inst}} \cdot \text{STR} \cdot \mathbf{q}_{\text{error}} = \kappa \sigma_i = \text{Kappa Att}_i \cdot \text{Res} \quad (6.4-200)$$

### 6.4.2.5 Interpolation of Attitudes

The  $\text{Level 1b}_{\text{Inst}} \cdot \text{STR} \cdot \mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}}$  data (also stored in  $\text{Level 1b.Eph}$ ) are interpolated to achieve the attitude at the required time-instants. As the time-instants are different for different processing chains, this process is repeated in the operational processors. The method for attitude interpolation is identical to the method used for ASM data, cf. Page 56, i.e. it is an iteratively reweighted least squares cubic B-spline approximation. Observe the introduction of leap seconds, cf. Section 4.6

First, the  $\mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}}$  data sequence is divided into segments with gaps not exceeding the maximum gap size as specified in  $(\text{CCDB.L1BP.STR.})\text{Spline\_Pars.Max\_Gap\_Size}$ . Then jumps in  $\mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}}$  caused by the sign ambiguity of quaternions are removed by changing signs on all four quaternion components on all  $\mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}}$  following a jump in one component of more than 0.9. The estimation of the B-spline functions to each data segment is susceptible to large deviations at the end points; this effect vanishes if the values at the end points are zero. This is accomplished by linear levelling of each data sequence by  $\mathbf{q}_{\text{CRF} \leftarrow \text{ICRF},i} - \mathbf{q}_{\text{Lin},i}$ ,

where  $q_{Lin,i}$  for a given data sequence  $y_0, y_1, \dots, y_{n-1}$  with corresponding time instants  $t_0, t_1, \dots, t_{n-1}$ , is found by:

$$q_{Lin,i} = \text{interpolate}_{Lin}([t_0, t_{n-1}], [y_0, y_{n-1}], t_i) \quad (6.2-175)$$

Within each leveled segment,  $q_{CRF \leftarrow ICRF,i} - q_{Lin,i}$ , a cubic B-spline  $f$  is estimated for each of the four quaternion components. The fit minimizes:

$$\chi^2 = \mathbf{d}^T \mathbf{C} \mathbf{d} + \lambda \mathbf{m}^T \Lambda \mathbf{m} \quad (6.4-210)$$

where:

$\mathbf{m}$  is the *model* vector, i.e. the parameters (spline coefficients) describing  $f$

$\mathbf{d} = (q_{CRF \leftarrow ICRF,i} - q_{Lin,i}) - f(\mathbf{m})$ , is the data residual vector of the current model – of component  $i$ .

$C$ :  $C_{ij} = w_i G_{ij}$ , where  $G$  is the design matrix (collocation matrix) for cubic B-splines for the given knots (internal spacing CCDB.L1BP.STR.Spline\_Pars.Knot\_Space, with additional knots in the ends), see Appendix D.  $\mathbf{w}$  are data weights computed as a combination of Huber weights [RD6],  $\mathbf{w}^{Huber}$ , and the data uncertainties, (Level 1b<sub>Inst</sub>.STR.) $\mathbf{q}_{error}$ . Let  $\alpha = \text{Spline\_Pars.Weight\_Huber}$ ,  $\beta = \text{Spline\_Pars.Weight\_ErrEst}$ , and  $\sigma_{q\_error} = \text{rms}(\mathbf{q}_{error})$ , then:

$$\mathbf{w} = \alpha \mathbf{w}^{Huber} + \beta 1.5 \sigma_{q\_error} / (\mathbf{q}_{error} + 0.1) + \max(\mathbf{1} - \alpha - \beta, \mathbf{0}) \quad (6.4-220)$$

$\mathbf{w}^{Huber}$  are determined iteratively. Initially,  $\mathbf{w}^{Huber} = \mathbf{1}$ .  $\mathbf{w}^{Huber}$  of the next iteration is computed by ( $i$  is the vector index):

$$w_i^{Huber} = \min\left(1, \frac{1.5}{|d_i / \sigma|}\right) \quad (6.4-230)$$

where  $\sigma$  is the root-mean-squared (rms) misfit of  $\mathbf{d}$  weighted by the weights  $\mathbf{w}$  of the preceding iteration:

$$\sigma = \sqrt{\frac{\sum (w_i d_i)^2}{\sum w_i}} \quad (6.4-240)$$

$\lambda = \text{Spline\_Pars.Acc\_Damping}$  is the damping factor of the second order time-derivatives.

$\Lambda$  is the block diagonal acceleration damping matrix, see Appendix D.

Equation (6.4-210) is solved by computing  $\mathbf{m} = (\mathbf{G}^T \mathbf{C} + \lambda \Lambda)^{-1} \mathbf{C}^T (q_{CRF \leftarrow ICRF,i} - q_{Lin,i})$ , and the whole procedure is iterated by recomputing  $\mathbf{w}$ ,  $C$  ( $G$  is constant),  $\mathbf{d}$  and  $\mathbf{m}$  until (though max 30 iterations):

$$\sqrt{\sum (m_i - m_i^{prev})^2} / \sqrt{\sum m_i^2} < 10^{-5} \quad (6.4-245)$$

The estimated B-splines are evaluated at UTC seconds and at the time-instants  $t_{out,VFM}$  and  $t_{out,EFI}$  – within the time-frames of the  $q_{CRF \leftarrow ICRF}$  data segments. After evaluation of the final four unleveled B-splines,  $f + q_{Lin,i}$ , the quaternions are normalized:

$$\mathbf{q}^{\text{norm}} = \mathbf{q} / \sqrt{\sum q_i^2} \quad (6.4-250)$$

This completes the  $\mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}}$  at all required time-instants.

The calculation of  $\dot{\mathbf{q}}_{\text{ITRF} \leftarrow \text{S/C}}$  is done using the time-derivative of  $\mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}}$  – the derivative of  $\mathbf{q}_{\text{ITRF} \leftarrow \text{ICRF}}$  is very small in comparison and is ignored; the other transformations are constant. The derivative of  $\mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}}$  is computed using the derivatives of the B-splines estimated above. The calculation of B-spline derivatives is given in Appendix D.

The error estimates of the combined quaternions (Level 1b<sub>Inst</sub>.STR. $\mathbf{q}_{\text{CRF} \leftarrow \text{ICRF}}$ ) are modified to form an intermediate error estimate,  $\mathbf{q}_{\text{error}}^{\text{new}}$ . This is the rms value of the original error estimate and the deviations from the four cubic B-splines,  $f_i$  evaluated at the  $t_{\text{out,STR}}$  time-instants (within the current segment):

$$\mathbf{q}_{\text{error}}^{\text{new}} = \sqrt{\left( \text{Level 1b}_{\text{Inst}} \cdot \mathbf{q}_{\text{error}}^2 + L_{\text{err}} \eta^2 \sum (q_i - q_{\text{Lin},i} - f_i)^2 / 4 \right) / (1 + L_{\text{err}})}$$

$$\mathbf{q}_{\text{error}}^{\text{new}} = \sqrt{\left( \text{Level 1b}_{\text{Inst}} \cdot \mathbf{q}_{\text{error}}^2 + L_{\text{err}} \eta^2 \sum (q_i - f_i)^2 / 4 \right) / (1 + L_{\text{err}})}$$

$$\mathbf{q}_{\text{error}}^{\text{new}} = \sqrt{\left( \text{Level 1b}_{\text{Inst}} \cdot \mathbf{q}_{\text{error}}^2 + L_{\text{err}} \eta^2 \sum (q_i - f_i)^2 / 4 \right) / (1 + L_{\text{err}})} \quad (6.4-260)$$

where  $\eta = 3600 \cdot 180 / \pi$  converts to arc-seconds, the unit of  $\mathbf{q}_{\text{error}}$ , and  $L_{\text{err}}$  is given in CCDB.L1BP.STR.Spline\_Pars.L\_error.

The error estimates of the interpolated attitudes are computed as the weighted average of the four nearest attitudes (sample numbers  $i$ ):

$$\mathbf{q}_{\text{error}}^{\text{out}} = \beta \frac{\sum \mathbf{q}_{\text{error},i}^{\text{new}} \text{wt}_i}{\sum \text{wt}_i} \quad (6.4-270)$$

where

$$\text{wt}_i = 1 / (|t - t_i| + 10^{-3}) \quad (6.4-280)$$

with  $t$  = time of the interpolated output quaternion and  $t_i$  = time of  $\mathbf{q}_{\text{error},i}^{\text{new}}$  – the time instants of the four nearest Level 1b<sub>Inst</sub> quaternions. (“+ 10<sup>-3</sup>” just to avoid division by zero)

$$\beta = \max\left(1, 21 / \sum \text{wt}_i\right) \quad (6.4-290)$$

$\beta$  increases the error estimate in case there are gaps in the Level 1b<sub>Inst</sub> data (sum over four nearest samples).



### 6.4.2.6 Flags

The values of  $\text{Flags}_q$  are given in Table 6-12 below. The higher flag values (“more serious”) take precedence over lower values, except for the “On-ground aberrational correction” condition (value 8) which is combined (added) to all other values, if applicable.

Value	Description
1,2,3	Lack of 1 or 2 attitudes of $\text{CHU}_i$ in 4 nearest STR samples, $i = \text{Value} (1,2,3)$
4,5,6	Lack of 3 or 4 attitudes of $\text{CHU}_i$ in 4 nearest STR samples, $i = \text{Value}-3$
8	On-ground aberrational correction of any attitude sample among 4 nearest STR samples. This value may be added to the value of all other applicable conditions.
16,17,18	$\text{CHU}_i$ obscured by bright object in 4 nearest STR samples, i.e. all 4 $\text{CHU}_i$ samples invalid or rejected from combination and marked with BBO flag, $i = \text{Value}-15$ , other CHU’s ok (i.e. these values override values 4-6 in case BBO is set)
19	2-4 attitudes based on $\text{CHU}_3$ and $\{\text{CHU}_1 \text{ or } \text{CHU}_2\}$ in 4 nearest STR samples, i.e. $\text{CHU}_3$ active in all 4 STR samples whereas $\text{CHU}_1$ and $\text{CHU}_2$ fail intermittently
20	2-4 attitudes based on $\text{CHU}_2$ and $\{\text{CHU}_1 \text{ or } \text{CHU}_3\}$ in 4 nearest STR samples, i.e. $\text{CHU}_2$ active in all 4 STR samples whereas $\text{CHU}_1$ and $\text{CHU}_3$ fail intermittently
21	2-4 attitudes based on $\text{CHU}_1$ and $\{\text{CHU}_2 \text{ or } \text{CHU}_3\}$ in 4 nearest STR samples, i.e. $\text{CHU}_1$ active in all 4 STR samples whereas $\text{CHU}_2$ and $\text{CHU}_3$ fail intermittently
22	3-4 attitudes based on two heads, and all three heads failing intermittently
32,33,34	1 or 2 attitudes based on $\text{CHU}_i$ alone (other CHU’s missing) in 4 nearest STR samples, $i = \text{Value}-31$
35	2 attitudes based on one, intermittent CHU alone in 4 nearest STR samples,
48	1 attitude sample missing among 4 nearest STR samples (data gap)
49	2 attitude samples missing among 4 nearest STR samples
50	3 or more attitude samples missing among 4 nearest STR samples
51,52,53	3 or 4 attitudes based on $\text{CHU}_i$ alone in 4 nearest STR samples, $i = \text{Value}-50$
54	3 or 4 attitudes based on one, intermittent CHU alone in 4 nearest STR samples
255	No STR data available for generating attitude information (large data gap)

Table 6-12  $\text{Flag}_q$  Values

### 6.4.2.7 Generation of Other Transformations and of $\mathbf{r}_{\text{VFM}}$

Level 1b.Eph. $\mathbf{q}_{\text{ITRF} \leftarrow \text{S/C}}$  at UTC seconds (1 Hz):

$$\text{Level 1b.Eph.}\mathbf{q}_{\text{ITRF} \leftarrow \text{S/C}} = \mathbf{STR\_q} \otimes \mathbf{q}^{-1}_{\text{ICRF} \leftarrow \text{ICRF}} \otimes \mathbf{q}_{\text{ITRF} \leftarrow \text{ICRF}} \quad (6.4-300)$$

where  $\mathbf{q}_{\text{ITRF} \leftarrow \text{ICRF}}$  is the quaternion representation of  $\mathbf{R}_{\text{ITRF} \leftarrow \text{ICRF}}$ , the ICRF  $\rightarrow$  ITRF transformation, see Section 8.4. Here and in the following, the CCDB.Structure prefix has been omitted.

$\mathbf{r}_{\text{VFM}}$  – position of the VFM sensor in ITRF frame – is computed at UTC seconds:

$$\text{Level 1b.Mag-L.}\mathbf{r}_{\text{VFM}} = \text{Level 1b.Eph.}\mathbf{r} + \mathbf{q}_{\text{ITRF}\leftarrow\text{S/C}} \times (\mathbf{VFM\_r} - \mathbf{r}_{\text{CoG}}) \quad (6.4-310)$$

with  $\mathbf{r}_{\text{CoG}}$  computed once per day – at noon – as specified in Appendix E page 150f..

Level 1b.Mag-H. $\mathbf{r}_{\text{VFM}}$  (at  $t_{\text{out,VFM}}$  time-instants) is computed by a cubic Lagrange interpolation of Level 1b.Mag-L. $\mathbf{r}_{\text{VFM}}$ .

$\mathbf{q}_{\text{NEC}\leftarrow\text{CRF}}$  at UTC seconds (Level 1b.Mag-L. $\mathbf{q}_{\text{NEC}\leftarrow\text{CRF}}$ ) and  $t_{\text{out,VFM}}$  time-instants (Level 1b.Mag-H. $\mathbf{q}_{\text{NEC}\leftarrow\text{CRF}}$ ):

$$\mathbf{q}_{\text{NEC}\leftarrow\text{CRF}} = \mathbf{q}^{-1}_{\text{CRF}\leftarrow\text{ICRF}} \otimes \mathbf{q}_{\text{ITRF}\leftarrow\text{ICRF}} \otimes \mathbf{q}_{\text{NEC}\leftarrow\text{ITRF}} \quad (6.4-330)$$

$\mathbf{q}_{\text{ITRF}\leftarrow\text{ICRF}}$  at  $t_{\text{out,VFM}}$  (50 Hz) may be computed by cubic Lagrange interpolation and re-normalization of  $\mathbf{q}_{\text{ITRF}\leftarrow\text{ICRF}}$  at 1 Hz.

$\mathbf{q}_{\text{NEC}\leftarrow\text{ITRF}}$  is the quaternion representation of the  $\mathbf{R}_{\text{NEC}\leftarrow\text{ITRF}}$  ITRF  $\rightarrow$  NEC( $\mathbf{r}_{\text{VFM}}$ ) transformation matrix, cf. Appendix C.

Computation of  $\mathbf{q}_{\text{NEC}\leftarrow\text{S/C}}$  at  $t_{\text{out,EFI}}$  time-instants is similar:

$$\mathbf{q}_{\text{NEC}\leftarrow\text{S/C}} = \mathbf{STR\_q} \otimes \mathbf{q}^{-1}_{\text{CRF}\leftarrow\text{ICRF}} \otimes \mathbf{q}_{\text{ITRF}\leftarrow\text{ICRF}} \otimes \mathbf{q}_{\text{NEC}\leftarrow\text{ITRF}} \quad (6.4-340)$$

Here,  $\mathbf{q}_{\text{NEC}\leftarrow\text{ITRF}}$  is also computed for the VFM sensor position – the tilt with respect to the  $\text{NEC}_{\text{EFI}}$  is insignificant w.r.t. EFI data.

## 7 Electric Field Instrument - EFI

### 7.1 Introduction

The Electric Field Instrument (EFI) consists of two components: the Langmuir Probe (LP) which determines the electron density, electron temperature and spacecraft potential and the Thermal Ion Imager (TII) which determines the ion density, the ion drift velocity and the electric field at the S/C front panel (in-flight). This section only refers to the LP algorithm as the TII processing is outside the scope of an automated Level 1b processor..

### 7.2 Input and Output Parameters

Inputs					
Elements		Description	Unit	Range	
L e v e l 1 a	L P	$I_1$	Current of ion-saturation, linear e- region, and retarded e- region in probe 1	eu	TBD
		$I_2$	Current of ion-saturation, linear e- region, and retarded e- region in probe 2	eu	TBD
		$D_1$	Derivative of ion-saturation, linear e- region, and retarded e- region in probe 1	eu	TBD
		$D_2$	Derivative of ion-saturation, linear e- region, and retarded e- region in probe 2	eu	TBD
		$I_{FP}$	16 values of currents from the face plate	eu	TBD
		Bias <sub>1</sub>	Bias voltage from tracking current on probe 1	eu	TBD
		Bias <sub>2</sub>	Bias voltage from tracking current on probe 2	eu	TBD
		U <sub>1,ret</sub>	Voltage bias retarded e- region probe1	eu	TBD
		U <sub>2,ret</sub>	Voltage bias retarded e- region probe2	eu	TBD
		Status	Possible overflows detected in processing		TBD
LP_Conf		LP configuration data			
L P - O f f	FP Curr,Off P1 Bias,Off Curr,Off Ref,ADC2 Ground P1 <sub>k</sub> P1 <sub>m</sub> P2 Bias,Off	FP <sub>Bias,Off</sub>	Face plate, bias offsets	eu	TBD
		FP <sub>Curr,Off</sub>	Face plate, current offsets	eu	TBD
		P1 <sub>Bias,Off</sub>	Probe 1, bias offsets	eu	TBD
		P1 <sub>Curr,Off</sub>	Probe 1, current offsets	eu	TBD
		P1 <sub>Ref,ADC2</sub>	Probe 1, reference ADC2	eu	TBD
		P1 <sub>Ground</sub>	Probe 1, ground	eu	TBD
		P1 <sub>k</sub>	Probe 1 slope, current offset	eu	TBD
		P1 <sub>m</sub>	Probe 1 constant, current offset	eu	TBD
P2 <sub>Bias,Off</sub>	Probe 2, bias offsets	eu	TBD		

Inputs						
Elements		Description	Unit	Range		
w e e p	L	$\mathbf{P2}_{\text{Curr,Off}}$	Probe 2, current offsets	eu	TBD	
		$\mathbf{P2}_{\text{Ref,ADC1}}$	Probe 2, reference ADC1	eu	TBD	
		$\mathbf{P2}_{\text{Ground}}$	Probe 2, ground	eu	TBD	
		$\mathbf{P2}_k$	Probe 2 slope, current offset	eu	TBD	
		$\mathbf{P2}_m$	Probe 2 constant, current offset	eu	TBD	
		T	LP electronics board temperature	°C	-30 – 40	
	P – S w e e p	L	$\mathbf{I}_{\text{sw},1}$	Sweep mode currents, probe 1	eu	TBD
			$\mathbf{I}_{\text{sw},2}$	Sweep mode currents, probe 2	eu	TBD
		–	dt	Step duration	s	0 – 0.1
		S	dU	Step height	V	±0.1
		w	$U_0$	Starting bias	V	±10
		e	$n_{\text{Inv}}$	Step at which sign of step height is inverted		0 – 255
		e	n	Number of sweep samples between steps		0 – 255
		p	Down <sub>CIC</sub>	Downsampling in FPGA CIC filter		0 – 255
Down <sub>CPU</sub>			Downsampling in CPU		0 – 255	
		Ctrl <sub>Mode</sub>	Control bits, mode		0 – 255	
	Ctrl <sub>HW</sub>	Control bits, hardware		0 – 255		
	AOCS.Mag_HK	Magneto-torquer coil currents	A	±10 <sup>2</sup>		
	AOCS.Thru_HK	Thruster activations		0/1		
	TCF.LP	Structure holding LP offset calibration parameters		[AD02]		
	Level 1b.Eph. $\mathbf{q}_{\text{ITRF} \leftarrow \text{S/C}}$	Transformation from S/C to ITRF frame		±1		
	Level 1b.Mag-L. $\mathbf{r}$	VFM position, ITRF	m	±10 <sup>7</sup>		
	Level 1b.Eph. $\mathbf{v}$	S/C velocity, ITRF	m/s	±10 <sup>4</sup>		
C C D B  C o n s t a	L P	...	Please refer to [AD09] for extensive list of instrument level processing parameters			
		Sun	Structure with look-up tables for visibility of probes, see [AD03]			
	C o n s t a	e	Electron charge	C	~1.60·10 <sup>-19</sup>	
		u	Atomic mass unit	kg	~1.66·10 <sup>-27</sup>	
		$m_e$	Electron mass	kg	~9.11·10 <sup>-31</sup>	
	$m_{\text{amu}}$	Mass of O <sup>+</sup> in amu	amu	15.999		

<b>Inputs</b>				
<b>Elements</b>		<b>Description</b>	<b>Unit</b>	<b>Range</b>
n	k2	Boltzmann constant	eV/K	$\sim 8.62 \cdot 10^{-25}$
t				
s				

<b>Outputs</b>			
<b>Elements</b>		<b>Description</b>	<b>Unit</b>
Level 1b Plasma	$U_{orbit}$	Spacecraft velocity in ITRF	m/s
	$N_e$	Plasma density	$cm^{-3}$
	$N_{e,error}$	Error of the plasma density estimate	$cm^{-3}$
	$T_e$	Plasma electron temperature	K
	$T_{e,error}$	Error of the electron temperature estimate	K
	$V_S$	Spacecraft potential	V
	$V_{s,error}$	Error of the spacecraft potential estimate	V
	$Flags_{LP}$	Flags indicating source of measurements	
	$Flags_{N_e}$	Flags related to $N_e$	
	$Flags_{T_e}$	Flags related to $T_e$	
	$Flags_{V_S}$	Flags related to $V_S$	
TCF.LP		LP offset calibration parameters	

## 7.3 Algorithms

The Langmuir Probe (LP) algorithms can be found in [AD09], cf. Appendix J

The S/C ephemeris and magnetic field vector data needed by the EFI algorithms are taken from various sources:

- The position is taken from Level 1b.Eph to co-locate plasma data with magnetic data. Interpolation in time is done using cubic Lagrange interpolation.
- The velocity is taken from Level 1b.Eph. The transformation of this from the ITRF to the NEC frame is given by

$$\mathbf{v}_{\text{NEC}} = \mathbf{R}_{\text{NEC} \leftarrow \text{ITRF}} \mathbf{v}_{\text{ITRF}} \quad (7.3-10)$$

where the transformation matrix  $\mathbf{R}_{\text{NEC} \leftarrow \text{ITRF}}$  is defined in Appendix C.

- The transformation from the S/C frame to the  $\text{NEC}_{\text{VFM}}$  frame is given by  $\mathbf{q}_{\text{ITRF} \leftarrow \text{S/C}} \otimes \mathbf{q}_{\text{NEC} \leftarrow \text{ITRF}}$ , see also Eq. (6.4-340) in Section 6.4.2.

### 7.3.1 Illumination of LP Probes

The illumination of the LP probes by the Sun generates a photo-electric effect which disturbs the LP measurements. To estimate whether the probes are sunlit or not, the position of the Sun is computed in the S/C frame and converted to spherical coordinates which are looked up in the CCDB.LP.Sun tables. The shadow of the Earth is also taken into account. Let

$\mathbf{r}$  = Level 1b.Mag-L. $\mathbf{r}$  – S/C position, ITRF

$\mathbf{r}_{\text{Sun}}$  = xl\_sun(t) – position of Sun, ITRF, see [RD12] for xl\_sun function

then if  $\mathbf{r} \cdot \mathbf{r}_{\text{Sun}} < 0$  and  $|\mathbf{r} \times \mathbf{r}_{\text{Sun}}| / |\mathbf{r}_{\text{Sun}}| < \text{CCDB.Constants.r}_E$  the S/C is in the shadow of the Earth and the probes are not illuminated. Otherwise compute position of the Sun in S/C frame:

$$\mathbf{s} = (\mathbf{q}_{\text{ITRF} \leftarrow \text{S/C}})^{-1} \times (\mathbf{r}_{\text{Sun}} - \mathbf{r}) \quad (7.3-20)$$

and convert  $\mathbf{s}$  to spherical coordinates ( $s$ ,  $\theta$ ,  $\varphi$ ). Make linear interpolations to determine illumination:

$$\text{illum}_k = \text{Interpol}_{\text{Lin}}(\text{CCDB.LP.Sun.Probe}_k, \theta, \varphi) \quad (7.3-20)$$

$k$  is the probe number (1 and 2),  $\text{Interpol}_{\text{Lin}}$  is defined in Appendix E.

## 8 Position and Acceleration

### 8.1 GPS Receiver - GPSR

#### 8.1.1 Introduction

Through the GPS antenna, the GPS receiver (GPSR) receives the signals from all of the antenna visible GPS satellites. The L1b processing corrects for known effects related to the Swarm instruments and satellite. The external errors e.g. due to the GPS segment are corrected in the orbit determination processing.

Eleven different input packet definitions from the ISP exists, these are

MDH	Measurement Data Header format
CAP	Carrier Phase data record
CAA	Carrier Amplitude data record
COP	Code Phase data record
GNA	GPS Nav Almanac data record
GNU	GPS Nav UTC and Ionosphere data record
MNS	Minimum Navigation Solution data record
IGC	IMT/GPST Correlation data record
GNE	GPS Nav Ephemeris data record
CS	Constellation Status record
AUX	Swarm Auxiliary data record

However, for the GPS algorithms processing only seven of them are used. The collection of the data record from the CAP, CAA, COP, and GNE has to be repeated N times for the N viewed satellites with a 0.1 Hz/1 Hz update frequency.

#### 8.1.2 Input and Output Parameters

Inputs				
Elements		Description	Unit	
L E V E L	M D H	UTC.DAYS	Number of days since 1 <sup>st</sup> of January 2000 in the UTC time base	UTC
		UTC.MSEC	Number of milliseconds of day	UTC
		UTC.USEC	Number of microseconds of milliseconds	UTC
		FE_TEMP	Instrument temperature	K
		0 D A	C A P	SV_ID
SIG	Signal type			
NCO_PHASE	Carrier Cycle count and Carrier NCO phase			m

<b>Inputs</b>				
<b>Elements</b>			<b>Description</b>	<b>Unit</b>
<b>T A</b>	<b>C A A</b>	SV_ID	Space vehicle identifier	
		SIG	Signal type	
		CNO	Carrier to noise power density ratio	dBHz
		AMPL	Carrier amplitude	dBW
	<b>C O P</b>	SV_ID	Space vehicle identifier	
		SIG	Signal type	
		CC	Number of code chips counts since start of GPS week	
		FP	Code phase fractional part	
	<b>M N S</b>	NMS	Navigation solution method	
		QUAL_IDX	Quality index	
		GDOP	Geometrical dilution of precision	
		POS_X	Position in WGS 84, x coordinate	m
		POS_Y	Position in WGS 84, y coordinate	m
		POS_Z	Position in WGS 84, z coordinate	m
		VEL_X	Velocity in WGS84, x velocity	m/s
		VEL_Y	Velocity in WGS84, y velocity	m/s
	<b>I G C</b>	IMT	Instrument measurement time representation of the synchronization time stamp.	Weeks and s
		GPST.SEC	GPS time representation of the synchronization time stamp.	S
		GPST.SUBSEC	Same as above, just in sub-seconds	s
	<b>G N E</b>	SV_ID	Space vehicle identifier	
		CRS	Amplitude of the Sine Harmonic Correction term to the orbit radius	m
		CRC	Amplitude of the Cosine Harmonic Correction term to the orbit radius	m
		CUS	Amplitude of the Sine Harmonic Correction term to the argument of latitude	rad
CUC		Amplitude of the Cosine Harmonic Correction term to the argument of	rad	



<b>Inputs</b>				
<b>Elements</b>		<b>Description</b>	<b>Unit</b>	
		latitude		
	CIS	Amplitude of the Sine Harmonic Correction term to the angle of inclination.	Rad	
	CIC	Amplitude of the Sine Harmonic Correction term to the angle of inclination.	Rad	
	DELTAN	Mean motion difference from computed value	rad/s	
	M0	Mean Anomaly at reference time	rad	
	E	Eccentricity		
	SQRTA	Square root of the semi major axis	$\sqrt{m}$	
	OMEGA0	Longitude of ascending node of orbit plane at weekly epoch	rad	
	I0	Inclination angle at reference time	rad	
	OMEGA	Argument of perigee	rad	
	OMEGADOT	Rate of right ascension	rad/s	
	TOE	Reference time ephemeris	s	
	IDOT	Rate of inclination angle	rad/s	
<b>C C D B  G P S R</b>	<b>GPSR_r</b>		Position of GPSR antenna phase centre in S/C coordinate system	m
	<b>GPSR_q</b>		Transformation from S/C frame to GPSR antenna reference frame	
	Ant.Patt_L1	Complex antenna radiation pattern representing the L1 C/A signal as function of azimuth and zenith angle	V	
	Ant.Patt_L2	Complex antenna radiation pattern representing the L2 P(Y) signal as function of azimuth and zenith angle	V	
	Ant.Code_CA	Antenna code range pattern of C/A on L1 as function of azimuth and zenith angle	m	
	Ant.Code_P1	Antenna code range pattern of P on L1 as function of azimuth and zenith angle	m	
	Ant.Code_P2	Antenna code range pattern of P on L2 as function of azimuth and zenith angle	m	

<b>Inputs</b>			
<b>Elements</b>		<b>Description</b>	<b>Unit</b>
	<b>ARP_r</b>	Position of antenna reference point (not used)	m
	Rx.Carr.L1_L2	Receiver carrier range bias for L2-L1 as a function of receiver temperature	m/C
	Rx.Code.P1_CA	Receiver code range bias for L1 P(Y) - L1 C/A as a function of receiver temperature	m/C
	Rx.Code.P2_CA	Receiver code range bias for L2 P - L1 C/A as a function of receiver temperature	m/C
	Rx.Cno.L1	Receiver C/No bias of L1 C/A as a function of receiver temperature	dB/C
	Rx.Cno.L2	Receiver C/No bias of L2 Y as a function of receiver temperature	dB/C
	Rx.No.L1	System noise density of receiver on L1 C/A as a function of receiver temperature	dB/C
	Rx.No.L2	System noise density of receiver on L2 Y as a function of receiver temperature	dB/C
	t_smooth	The smoothing interval half width of the temperature smoothing	
	GPS.L1	GPS L1 frequency, $f_1$	Hz
	GPS.L2	GPS L2 frequency, $f_2$	Hz
	ResIF_L1	Residual IF at L1	
	ResIF_L2	Residual IF at L2	
	F_CA	CA cod chip rate	Hz
	F_P	P code chip rate	Hz
	B_CA	CA code loop bandwidth	Hz
	d_CA	CA early late spacing	
	T_CA	CA coherent integration time	s
	Constants.c	Speed of light	m/s
<b>STR output</b>	$\mathbf{q}_{ITRF \leftarrow S/C}$	Quaternion S/C attitude; output from the STR Instrument level processing	

<b>Outputs</b>
RINEX Observation File <i>Version 3.0</i>
RINEX Navigation File <i>Version 3.0</i>

For further definition of the RINEX format, see RD 11. The decision on the version of the RINEX format will be done in the requirements consolidation phase. The RINEX Observation and Navigation files are roughly organized as follows:

<b>R I N E X  O b s e r v a t i o n s</b>	<i>H e a d</i>	Ant_bore	Antenna boresight	
		CoG_sat	Center of Gravity of the LEO satellite	m
		Ant_pos	Antenna position	m
	<i>D a t a</i>	LEO_pos	Approximate LEO position in WGS84	m
		Obs_type	Type of observations	
		Obs_scalef	Observation scale factor	
		AUX	Auxiliary information	
		t_epoch	Time for each epoch	GPS
		N_sat	Number of measured GPS satellites	
		L1	Phase information on the L1 frequency	Cycle
		L2	Phase information on the L2 frequency	Cycle
		C1	Pseudo-range of the C/A code on the L1 frequency	m
		P1	Pseudo-range of the P-code on the L1 frequency	m
	<i>R e c o r d</i>	P2	Pseudo-range of the P-code on the L2 frequency	m
		S1	Raw signal strength on L1 – or SNR	
		S2	Raw signal strength on L2 – or SNR	
		CH	Receiver channel number	
		L_lock	Loss of lock indication	
		P_signal	Signal strength (quality)	Watt
<b>N A V I G A.</b>	<i>H e a d</i>	ION_par	Ionospheric parameters from almanac	
		UTC_par	UTC calculation parameters	
		AUX	Auxiliary data information	
<i>D a t a</i>	t_epoch	Time for each epoch	GPS	
	N_sat	Number of measured GPS satellites		
	EPH	Ephemeris data of observed GPS satellites		

### 8.1.3 GPSR Algorithm

The processing flow of algorithms is separated into three major steps:

- *L0 processing*, which comprises reading and checking of the raw binary level 0 instrument data.

- *L1a processing*, which comprises reformatting as well as interpretation and unit conversions of the L0 ISPs. The result is stored as L1a products.
- *L1b processing*, which comprises correction of the L1a data corresponding to the characterized distortions comprised in the Instrument Characterization Data Base (ICDB) which will be a part of the CCDB. The result is stored as RINEX 3.00 formatted ASCII data.

The overall data flow from L0 to L1b is show in Figure 8-1. An overview of the sequential walk-through of the data flow will be given below. For details a section number referring to AD11 is given and the associated algorithms are given in brackets in the sequential walk-through.

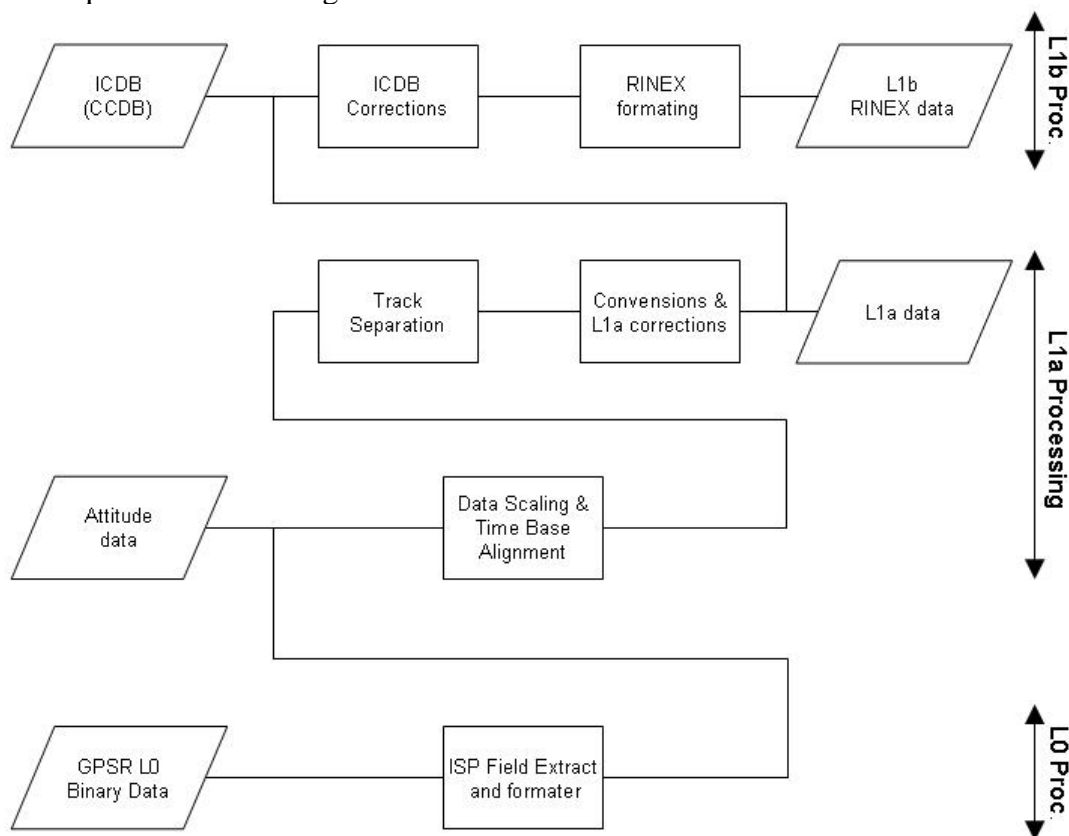


Figure 8-1 Level 0 to Level 1b processing flow for the GPSR.

1. First the level 0 to level 1a data processing, as defined in chapter 5 of this document, shall be performed. This includes also the following specific GPSR processing tasks.

- ISP are scaled into floating point values (§6.1.1)
- The instrument measurement time (IMT) shall be aligned to the GPS time (GPST) [GPSR§6.1.2]<sup>14</sup> and the data and time shall be stamped in all three time bases: UTC, GPST, and IMT.

<sup>14</sup> GPSR§x.y.z provides reference to section x.y.z of the GPSR L1b Algorithms Definition [AD11]

- As an additional input the spacecraft attitude is needed. For this the result of the STR processing, chapter 6.4, i.e. the rotation of ITRF to S/C frame, shall be used. The algorithms defined by [AD11] (Section 6.2.1.2) need the attitude defined by yaw, roll and pitch at certain time stamps. Therefore the quaternion rotation from STR processing shall be transformed to roll, pitch, and yaw angles via the orbital reference frame. Let  $M_{\text{itr}f2\text{SC}}$  be the rotation corresponding to  $\mathbf{q}_{\text{ITRF-S/C}}$ , and  $\mathbf{R} = M_{\text{itr}f2\text{SC}} \times M_{\text{or}f2\text{wgs}}$  (eq-11 of [AD\_11]), then

$$\begin{aligned} \text{pitch} &= \text{asin}(\mathbf{R}_{3,1}) \\ \text{roll} &= -\text{atan2}(\mathbf{R}_{3,2}, \mathbf{R}_{3,3}) \\ \text{yaw} &= -\text{atan2}(\mathbf{R}_{2,1}, \mathbf{R}_{1,1}) \end{aligned} \quad (8.1-10)$$

These are then interpolated to the required time stamps [GPSR§6.1.3].

Note: In the requirements consolidation phase it should be considered whether the processing can be simplified by calculation of the quaternion already for the required time stamps and using quaternions for the further GPSR processing in [GPSR§6.2.1.2].

- Temperature data shall be smoothed (e.g. using a triangular weighting of the temperature measurements) [GPSR§6.2.2.1]
- Before further processing the GPSR data should be remapped to a track oriented mapping. This “track separation” parses through the data from carrier phase, carrier amplitude, carrier amplitude C/No, and the code phase, and separates the data into tracks.

The tracks are associated by the following criteria [GPSR§6.1.4]:

- ◇ Data from the same GPS satellite, i.e same PRN number
  - ◇ Data that is measured on the same multi frequency channel (MF\_chn)
  - ◇ The data gap between samples is below 100s
- Provision of GPS Ephemeris Data. The GPS Ephemeris are to be taken from the related level 0 product, when available or alternatively from auxiliary data files to be downloaded from IGS

Next step are the I1a conversion and corrections. This conversion generates the level 1a data based on the demultiplexed, scaled, time correlated and track separated level 0 data. Further inputs to this processing are the spacecraft attitude information and the GPS Ephemeris Data. The I1a conversion and correction shall cover:

- GPS PVT: Position, velocity and time at the time of transmission shall be calculated for each GPS, ref. [GPSR§6.2.1.1]
- The azimuth, elevation and range to the LEO satellite shall be calculated for each GPS satellite based on the GPS PVT, the LEO PVT and the attitude information, ref. [GPSR§6.2.1.2].
- The data shall be merged with further data extracted from the level 0 products to GPS EPVT and LEO EPVT data sets that will be available as level 1a products. The detailed contents is to be defined during the design phase, for a first definition of the contents ref. [GPSR§5.2.1]

- **Carrier range:** Carrier range, in units of meters, shall be determined from the accumulated carrier phase reported by the instrument. It shall be corrected for downconversion scheme of the instrument and demodulated for the residual IFs, given in the CCDB ref. [GPSR§6.2.3.1]. Additionally, it is aligned to the GPS time (provided by the time correlation package), as a time reference. This allows to have the measurements aligned to the GPS time, instead of the IMT time, and as a result of this, the epochs in the final Rinex file are aligned to the integer second and do not drift with time.
- **Code range:** Code range, in units of meters, shall be determined from the chip counts and the fractional code phase reported by the instrument. Code range shall be reported for the C/A and P(Y) code on L1 and the P(Y) code on L2 depending on the tracking status of the instrument. Code range shall be reported at the selected code phase sample rate for the instrument.
- **Code transmission time:** The transmit time of the individual GPS signals shall be determined. The algorithm requires the code phase measurement and the GPS receiver time provided in the time correlation packages. No ambiguity resolution is applied.
- **Pseudo Range:** The code Pseudo Range (PR), in units of meters, shall be determined as the difference between the GPS time (provided by the time correlation package) and the GPS transmission time determined from the code phase samples. For details ref. [GPSR§6.2.3.3]
- **Cycle slip detection and correction:** All observations shall be checked and the time intervals that are corrupted by cycle slips shall be identified. If possible the cycle slips shall be repaired. The reported data shall be tagged with information on cycle slip detection and correction. For details refer to section 8.2.4 (MAUPRP section).

As a further step the pseudo ranges are corrected by an ad-hoc hardware bias, which has been calibrated to align the estimated clocks to the GPS time scale. The value corrected is included in the header of the Rinex files. No other calibration is performed to keep the data as raw as possible.

The final step is the RINEX formatting. The output from the former algorithms steps together with further data extracted from the level 0 products shall be formatted into the RINEX 3.00 format.

## 8.2 Orbit Determination

### 8.2.1 Introduction

The Medium Accurate Orbit Determination MOD shall run automatically in a “black-box” solution, and fed with the output data from the GPSR (RINEX Observation files), IGS’ GPS orbits, and the Earth orientation parameters.

Because of the very high complexness of the parameter estimation of the orbit determination the algorithms are not trivial and are acquired after many years of experience. An existing “plug-and-play” orbit determination software does **not** exist. Two possibilities can be considered:

1. Perform some kind of smoothing or filtering of the already onboard navigational solution performed by the GPSR. These algorithms could be implemented in the L1b processor.
2. Using already developed orbit modeling software.

#### 8.2.1.1 Filtering/Smoothing of Navigational Solution

GPS expert’s opinion is that filtering or smoothing of the navigational solution over an arc will achieve higher accuracy than 20m RMS. However, a problem will be to reach the accuracy of 1m RMS in all three directions. To smooth an orbit solution with accuracy of 20 RMS to 1m meter is a very challenging task, and perhaps not possible. By doing a smoothing the used algorithms can be included in the L1b processing and need no external and perhaps manual control.

#### 8.2.1.2 Existing POD Software

Several POD software packages exist and some has been considered in consultation with experienced POD experts to be sure to deliver an orbit determination in an automatic processing of 1m accuracy for the MOD or even a manual processing of 10 cm accuracy for the POD. After many consultations and investigations of some of the software-packages we have decided to choose the Bernese 5.0 software. By the use of Bernese, it is guaranteed that the ~1m RMS requirement of the MOD will be reached. Some disadvantages might be:

- It is a more or less commercial package, and has to be paid – Contractor B-license has to be obtained through the Leica Geosystems company
- According to a statement by M. Rothacher (GFZ) (formerly programmer of parts of the software) Bernese is **not** documented according to ESA software standards and the algorithms are only sparsely documented.
- Primarily a GPS processing software and no commercial scripts for POD or MOD are available – hence Perl module scripts has to be written
- Highly sophisticated software that need experience to handle and set up

### 8.2.2 Input and Output Parameters

Inputs			
Elements	Description	Unit	Accuracy
	Header		
	This part is roughly organized as mentioned in Section 8.1.2. This is		

<b>RINEX Observation</b>		stored only once per file.		
	Data record	This part is roughly organized as mentioned in Section 8.1.2. The data record is repeated N times for N satellites in view for each epoch.		
<b>STR output</b>	$\mathbf{q}_{ITRF \leftarrow S/C}$	Quaternion S/C attitude; output from the STR L1b processing (optional: Bernese may be run with or without attitude data)		
<b>External Data</b>	sp3	IGS GPS Satellite Ephemeris, minimum 'rapid' resolution is needed.		<5cm
	clk	IGS high rate GPS satellite clocks and corrections. Same resolution as sp3.	s	<1 $\mu$ s
	erp	Earth rotation parameters, used for transformation between ITRF/ICRF. Same resolution as sp3.	mas mas/day	<0.1mas <0.2mas/day
	SAT_YYYY.CRX SATELLIT.I01	GPS satellite problem file Satellite information file		
	CONF	Bernese configuration files, e.g. leap seconds, phase center corrections, datum information, satellite information files. <sup>15</sup>		

<b>Outputs</b>			
<b>Elements</b>	<b>Description</b>	<b>Unit</b>	<b>Range</b>
<b>r_ITRF</b>	Position vector in ITRF system	m	~1m
<b>v_ITRF</b>	Velocity vector in ITRF system	m/s	~1cm/s

### 8.2.3 Orbit Determination Processing

The orbit determination is organized as sketched in Figure 8-2. Below is a cursory description of the individual processing steps necessary. For details, e.g. on Bernese directory structure and input/output files and formats, please see AD12.

<sup>15</sup> All Swarm specific configuration files to be delivered by DNSC, all other auxiliary data files are included in Bernese distribution. GPSUTC file must be updated in case of future introduction of leap seconds.



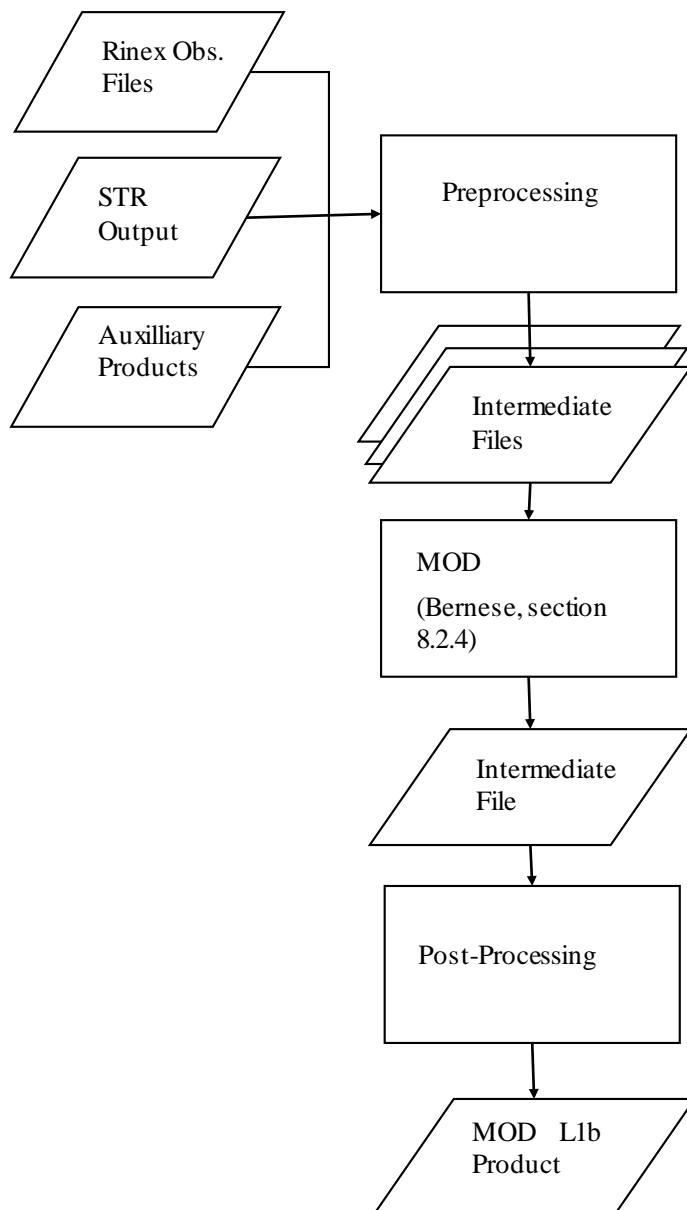


Figure 8-2 Orbit Determination Processing

**Pre-processing:**

Basically, pre-processing consists of downloading the necessary files, renaming according to Bernese standard and placing them in the relevant Bernese campaign sub-directories. Some possible exceptions:

1. Rinex observation files: if given as sub-daily (hourly) files these should be concatenated. This may be done in pre-processing or by the Bernese software.
2. Attitude data: the Bernese software may read attitude quaternions in Champ or Jason formats, or as native Bernese rotation matrix file. The attitude file should be reformatted to one of these options or an update of the Bernese READATT subroutine should be written to accept Swarm attitude files.

### MOD processing:

The Medium accuracy Orbit Determination is carried out using the Bernese 5.0 software package, please see section 8.2.4. The output produced by Bernese is a precise orbit file SWNnyydd.pre containing

1. Header section (22 lines) containing processing information (time of first epoch, sample frequency, satellites processed).
2. Data section containing 3 lines for each epoch (each second):
  - a. Time of epoch
  - b. Satellite position
  - c. Satellite velocity

### 8.2.4 Bernese Software Orbit Determination Steps

In Figure 8-3 the overall processing of both the MOD and the Final POD are sketched. Bernese is a Fortran based software invented by the Astronomical Institute at the University of Bern, Switzerland. The current Version 5.0 consists of a graphical user interface (GUI) with a front-end for automatization called the Bernese Processing Engine (BPE). It can be installed on UNIX/Linux or Windows platforms.

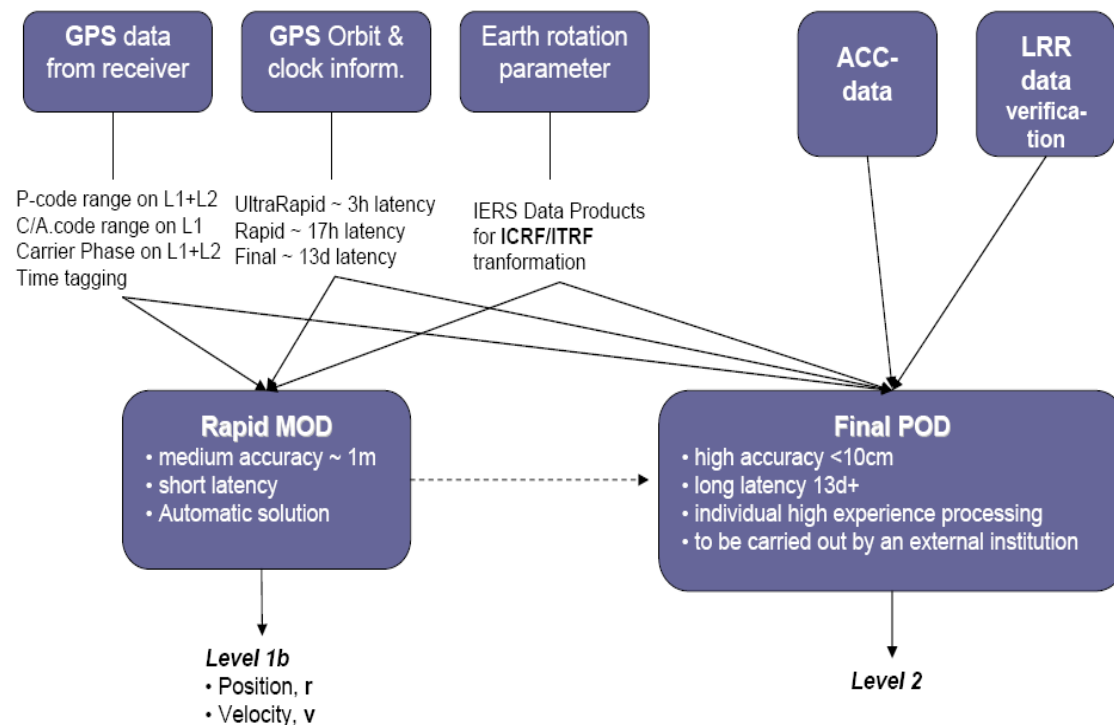


Figure 8-3 The overall processing of the orbit determination, both for the MOD and the final POD.

Several different programs in Bernese have to be run in order to receive the final position and velocity of the orbit determination. The overall dataflow is show in Figure 8-4. In the top the IGS GPS orbits and clock, the ERP, the RINEX observation files and some Bernese specific auxiliary data are show as input data. The rectangular boxes with hard edges shows the individual programs that are executed and the boxes with round corners are the output from

the programs. The orbit determination is performed in three steps, indicated by three dashed boxes:

- 1) the orbit part, where the GPS satellites orbits are estimated using a dynamical approach,
- 2) the pre processing, where an a priori orbit are generated,
- 3) the reduced-dynamic POD approach, where all the needed parameters are estimated and stochastic pulses from the GPS-estimated parameters are used as further input into the dynamic processing.

#### 8.2.4.1 The Preliminary Processing

In the preliminary processing flow the main task is to convert the input data to a format readable for the Bernese Software.

#### 8.2.4.2 The Orbit Part

The main task for the orbit part is to generate the GPS orbits. For this purpose the program ORBGEN which uses a dynamical process is used. Since this program is also used for the generation of a priori orbits and the final MOD orbit, the algorithmic background for this program is further described:

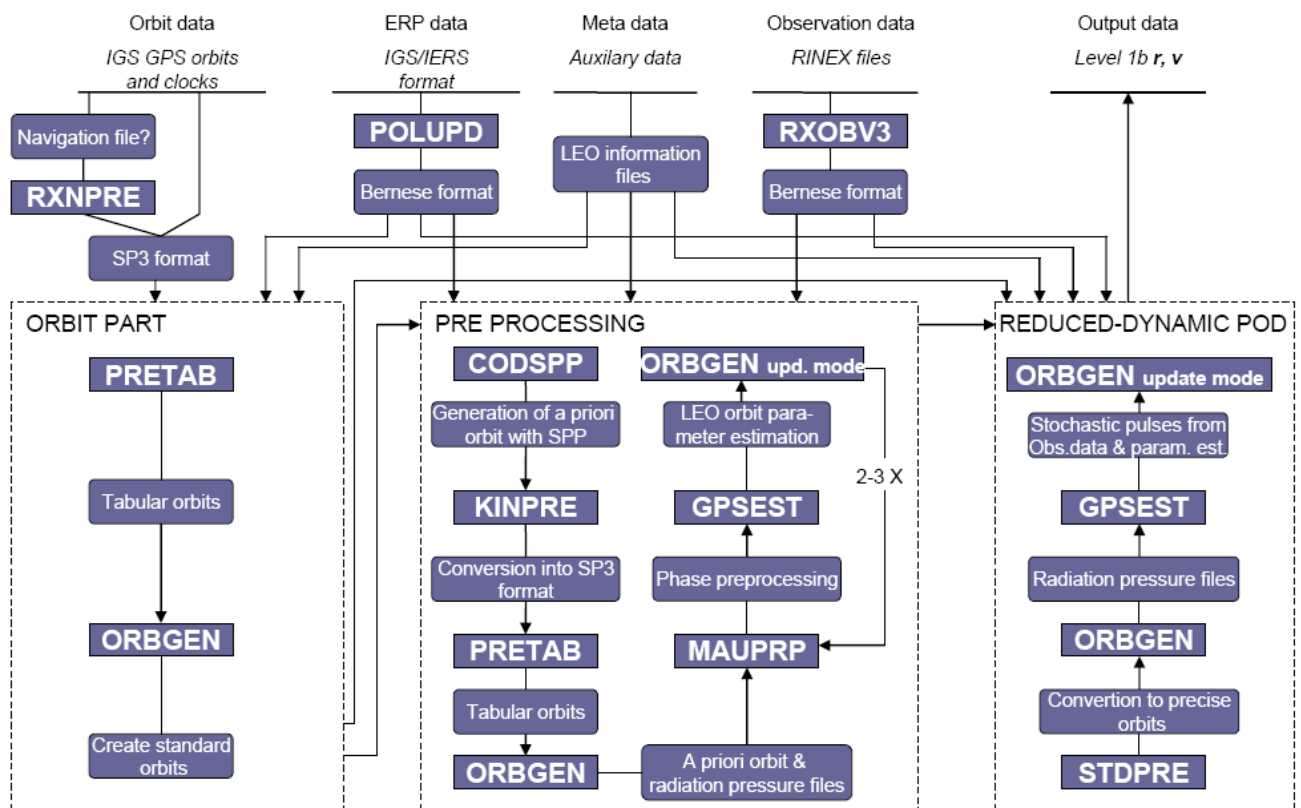


Figure 8-4 The overall processing of the MOD with the Bernese Software.

#### ORBGEN

The ORBGEN program integrates numerically the equation of motion, which basically is the main equation for estimation of a position and d velocity of a satellite. Each GPS orbit (each arc of the orbit) is a solution of the equation of motion given by

$$\begin{aligned} \ddot{\vec{r}} &= -GM \frac{\vec{r}}{r^3} + \delta\vec{f}(t; \vec{r}, \dot{\vec{r}}, p_1, p_2, \dots, p_m), \\ \vec{r}(t_0) &= \vec{r}(s_i(t_0)), \\ \dot{\vec{r}}(t_0) &= \dot{\vec{r}}(s_i(t_0)), \end{aligned} \tag{8.2-10}$$

where  $G$  is the universal constant of gravitation,  $\delta\vec{f}$  is the perturbation accelerations,  $\vec{r}$  the geocentric position vector of the satellite,  $\dot{\vec{r}}$  is the first time derivative,  $\ddot{\vec{r}}$  its second time derivative of the satellite position,  $p_i$ ,  $i = 1, 2, \dots, m$  which is the unknown dynamical parameters of the orbit, and  $s_i$  is any one of the six classical Keplerian elements that uniquely describe the satellite orbit.

The force model used within Bernese includes the Earth's potential up to a selectable degree and order, the gravitational attractions of Sun and Moon as well as of the planets Jupiter, Venus, and Mars, the elastic Earth tidal corrections according to IERS 1996 conventions, pole tide, ocean tides, and general relativistic corrections.

When determining an orbit (ORBGEN are used both for the final determination of the reduced-dynamic POD and for the orbit determination for the GPS orbits, see Figure 8-4) first the six parameters defining the position and velocity at the initial epoch  $t_0$  of the arc has to be specified – these are called the inertial conditions. These are done in ORBGEN by use of the six osculating Keplerian orbital elements. The dynamical parameters used in Bernese are attributed into the radiation pressure model. The acceleration of the radiation pressure model,  $\vec{a}_{rp}$ , is modelled in the following way:

$$\vec{a}_{rp} = \vec{a}_{MODEL} + \vec{a}_D + \vec{a}_Y + \vec{a}_X, \tag{8.2-20}$$

where  $\vec{a}_{MODEL}$  is an a priori radiation pressure model, and

$$\begin{aligned} \vec{a}_D &= (a_{D0} + a_{DC} \cos u + a_{DS} \sin u) \vec{e}_D \\ \vec{a}_Y &= (a_{Y0} + a_{YC} \cos u + a_{YS} \sin u) \vec{e}_Y \\ \vec{a}_X &= (a_{X0} + a_{XC} \cos u + a_{XS} \sin u) \vec{e}_X \end{aligned} \tag{8.2-30}$$

where  $a_{D0}, a_{DC}, a_{DS}, a_{Y0}, a_{YC}, a_{YS}, a_{X0}, a_{XC}, a_{XS}$  are the nine parameters of the radiation pressure model Bernese has to estimate,  $\vec{e}_D, \vec{e}_Y, \vec{e}_X$  are the three unit vectors 1) in the Sun direction, 2) along the spacecraft's solar-panel axis assuming nominal satellite attitude, and 3) that completes a right-handed orthogonal system, and finally  $u$  is the argument of latitude at time  $t$ <sup>16</sup>. The a priori model is known for the Block I and Block II GPS satellites, but not for Swarm. Using ORBGEN only for the LEO orbit determination, the  $\vec{a}_{MODEL}$  term is set to zero.

Hence the minimum number of parameters to be estimated per arc is  $n = 6 + 9 = 15$ . However, this is only done in the dynamical process. In the reduced-dynamic approach, the orbit is additionally parameterized with so-called pseudo-stochastic parameters, characterizing instantaneous velocity changes at user-determined epochs in user-determined direction (typically all three directions). The attribute stochastic is justified because usually a priori

---

<sup>16</sup>  $u$  is the sum of the two Kepler elements: the mean anomaly,  $M$ , and the argument of the orbits perigee,  $\omega$ , for a nominal satellite attitude.

weights (i.e., variances) are associated with these parameters. The attribute ‘pseudo’ is used because we are not allowing the orbits to adjust themselves continuously at every measurement epoch.

For the solution of the dynamical equation ORBGEN has to compute the partial derivatives of the position and velocity vector with respect to all orbit parameters. Because of the perturbation acceleration term  $\delta\vec{f}$  no analytical solution can be found for the partial derivatives. Since for longer arcs an analytical approximation is not sufficient in all cases, all partials computed by ORBGEN are found using numerical integration method. The procedure is to derive one set of differential equations, called variational equations, and one set of initial conditions, for each parameter. Then the resulting initial value problem will be solved by numerical integration.

If we write the dynamical equation as

$$\ddot{\vec{r}} = -GM \frac{\vec{r}}{r^3} + \delta\vec{f}(t; \vec{r}, \dot{\vec{r}}, p_1, p_2, \dots, p_m) = \vec{f}(t; \vec{r}, \dot{\vec{r}}, p_1, p_2, \dots, p_m) \quad (8.2-40)$$

then the variational equations can be derived by taking the derivative of the equation above with respect to the parameters,  $p$ . This gives the following initial value problem (variational equation and associated initial conditions):

$$\ddot{\vec{r}}_p = \mathbf{A}\vec{r}_p + \vec{f}_p \quad (8.2-50)$$

$$\vec{r}_{0,p} = \vec{r}_p(s_i(t_0)) \quad (8.2-60)$$

$$\vec{v}_{0,p} = \vec{v}_p(s_i(t_0))'$$

where it is assumed that there are no velocity-dependent forces.  $\mathbf{A}$  is a  $3 \times 3$  matrix with  $A_{p,ik} = \partial\vec{f}_i / \partial\vec{r}_k$  for element  $i, k$  and  $\vec{f}_p$  is the explicit derivative of  $\vec{f}$  with respect to the parameter  $p$  (equal to zero for osculating elements). The initial conditions are zero for the dynamical parameters.

In the orbit part, ORBGEN creates standard GPS orbits using a dynamical approach. These orbits serve as input to almost every program used in the further MOD processing.

### 8.2.4.3 The Pre Processing

In the pre processing the RINEX observation files are processed. Together with the processed GPS orbits and the Bernese internal auxiliary data, all these data serves as input in the pre processing for generation of good a priori LEO orbits. It will also be used for the final reduced dynamic MOD processing. The main programs in the pre processing are, CODSP, ORBGEN, MAUPRP, and GPSEST.

In the pre processing the GPS observations are used as input and in basically every program the GPS observation equations are used. Four basic GPS observation equations are currently used. Two for the code pseudo-ranges and two for the phase pseudo-ranges – each pseudo-range can use two frequencies  $F$ , both  $L_1$  and  $L_2$  with frequency  $f_1$  and  $f_2$ .

If the signal reception time (GPS time system) is called  $t$  and the reading of the receiver clock at signal reception is called  $t_k$ , then the error of the receiver clock (with respect to GPS time) is given by

$$\delta_k = t_k - t. \quad (8.2-70)$$

Further we define the geometrical distance between GPS satellite  $i$ , at the emission time  $t - \tau$ , and receiver  $k$ , at reception time  $t$ , as

$$\rho_k^i = |\vec{\mathbf{r}}_k(t) - \vec{\mathbf{r}}^i(t - \tau)| = c\tau, \quad (8.2-80)$$

with  $\tau$  being the signal travel time from GPS satellite to receiver,  $c$  the velocity of light,  $\vec{\mathbf{r}}_k$  the position of the LEO satellite, and  $\vec{\mathbf{r}}^i$  the position of the  $i$ 'th GPS satellite.

Using the known codes modulated onto the GPS carriers, the receiver measures the pseudo-range

$$P_k^i = c((t + \delta_k) - (t - \tau + \delta^i)), \quad (8.2-90)$$

with  $\delta^i$  being the error of the GPS satellite clocks at signal emission time  $t - \tau$ .  $P_k^i$  is called the pseudo-range because it is biased by satellite and receiver clock errors. Writing Eqn. (4) in terms of geometrical distance for each frequency the position of the LEO satellite can be introduced

$$P_{Fk}^i = \rho_k^i + c\delta_k - c\delta^i. \quad (8.2-100)$$

To achieve phase pseudo-ranges the GPS receiver measures the difference between two phases. The basic form of the observation equation is

$$\psi_{Fk}^i(t) = \phi_{Fk}^i(t) - \phi_F^i(t - \tau) + n_{Fk}^i, \quad (8.2-110)$$

where  $\psi_{Fk}^i(t)$  is the phase measurement (in cycles) at epoch  $t$  and frequency  $F$ ,  $\phi_{Fk}^i(t)$  is the phase generated by the receiver oscillator,  $\phi_F^i(t - \tau)$  is the phase of the carrier at emission time, and  $n_{Fk}^i$  is an unknown integer number of cycles, the so-called initial phase ambiguity.

Using a Taylor series expansion of first order on the  $\phi_F^i(t - \tau)$  term and taken the receiver and GPS satellite clock errors into account, the phase observation equation can be written as

$$\psi_{Fk}^i(t) = (\delta_k - \delta^i)f_F + \mathcal{T}_F + n_{Fk}^i. \quad (8.2-120)$$

By multiplying this equation by the wavelength  $\lambda_F$ , we receive the phase observation in meters

$$L_{Fk}^i = \rho_k^i + c\delta_k - c\delta^i + \lambda_F n_{Fk}^i. \quad (8.2-130)$$

The phase measurement and the code pseudo-ranges are affected by both systematic and random error. All relevant systematic errors should carefully be modeled in the used GPS software, but traditionally two of them are included in the observation equations, namely the tropospheric refraction  $\Delta\rho_k^i$  and ionospheric refraction  $I_k^i$ . Taken this into account, the four basic observation equations can be written as

$$P_{1k}^i = \rho_k^i + c\delta_k - c\delta^i + I_k^i + \Delta\rho_k^i \quad (8.2-140a)$$

$$P_{2k}^i = \rho_k^i + c\delta_k - c\delta^i + \frac{f_1^2}{f_2^2} I_k^i + \Delta\rho_k^i \quad (8.2-140b)$$

$$L_{1k}^i = \rho_k^i + c\delta_k - c\delta^i - I_k^i + \Delta\rho_k^i + \lambda_1 n_{1k}^i \quad (8.2-140c)$$

$$L_{2k}^i = \rho_k^i + c\delta_k - c\delta^i - \frac{f_1^2}{f_2^2} I_k^i + \Delta\rho_k^i + \lambda_2 n_{2k}^i \quad (8.2-140d)$$

Due to the frequency dependence the ionospheric refraction, the  $L_2$  is written as the squared fraction between the two frequencies  $f_1$  and  $f_2$ . This is the four basic GPS observables that are also used in the Bernese software.

For LEO orbit modeling the tropospheric refraction is automatically disabled, since the received signal is out of the troposphere. To eliminate the ionospheric refraction one can take advantages of the dual frequencies the GPSR in Swarm delivers. It is possible to form a linear combination between the two frequencies called the Ionosphere-Free linear combination,  $L_3$ , which is defined for the phase measurement as

$$L_3 = \frac{1}{f_1^2 - f_2^2} (f_1^2 L_1 - f_2^2 L_2). \quad (8.2-150)$$

The same is true for the corresponding combination of code measurements. By this linear combination the ionospheric refraction is canceled.

### CODSPP

The main task of CODSPP is to compute the receiver clock corrections. The receiver clock error has to be known with accuracy better than  $1\mu\text{s}$ . It would be possible to introduce the receiver clock error as unknown parameters during the final least-squares adjustment in program GPSEST, but this would increase the number of parameters considerably. Fortunately, it is possible to compute the clock correction, together with the coordinates of the LEO, a priori with sufficient accuracy ( $< 1\mu\text{s}$ ) using code measurements. This procedure is called Single Point Positioning (SPP).

The linearised observation equation to compute the receiver clock correction and a priori position for every epoch in CODSPP is given by

$$l_i = \frac{-1}{\rho_k^i} (\bar{\mathbf{r}}^i - \bar{\mathbf{r}}_0) \bullet (\bar{\mathbf{r}}_k - \bar{\mathbf{r}}_0) - \Delta t + \bar{v}_k, \quad (8.2-160)$$

where  $\bar{\mathbf{r}}_0$  the initial condition,  $\Delta t$  is the receiver clock error in length, i.e.  $\Delta t = c\delta_k$ , and  $\bar{v}_k$  is the residual. The term,  $l_i$ , is the so-called observed minus computed pseudorange (O-C) and is calculated for each observation of an epoch introducing the GPS satellite orbits, GPS satellite clocks, and the a priori position of the LEO. Based on these O-C values the receiver clock correction and the positions for the LEO for this particular epoch is obtained by majority voting. After correcting the O-C values for all observations for the receiver clock corrections the residual are analyzed for outliers. If an observation of an epoch exceeds a specified threshold the current epoch is marked both in phase and code observations.

The above equation can be written in general terms of matrices and is solved with least squares adjustment. The coefficients of the equation are computed using the initial coordinate vector. The a priori coordinates may not be accurate enough or even not known and for that reason the equation are solved iteratively. Note also that the equation has four unknowns and can then only be solved with a minimum of four GPS satellites available for the particular epoch. If fewer than four satellites are available CODSPP (and also all further processing)

cannot compute the SPP and clock correction and therefore this particular epoch will be deleted.

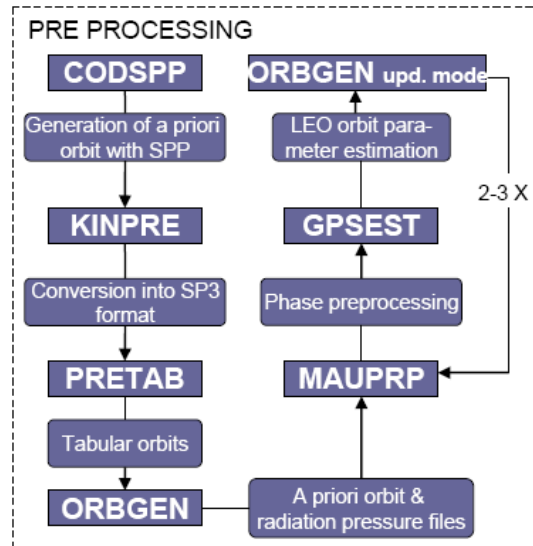


Figure 8-5 The Pre Processing of the Bernese orbit determination.

### ORBGEN

The important output from this process here is the generation of the radiation pressure model for the LEO satellite. For this purpose the orbit integration has to be based on a recent gravity model (EIGEN2) including the Colombo parameters with a degree and order of minimum 70. The integration step has to be selected to 1 min in order to achieve the lowest RMS values.

### MAUPRP

The following tasks are accomplished by MAUPRP:

- (1) Marking of observations due to the following reasons in order to exclude them from the preprocessing and the data analysis:
  - observations at low satellite elevation,
  - epochs with unpaired observations ( $L_1$  without  $L_2$  or vice versa),
  - small pieces of observations.
- (2) Perform a non-parametric screening to identify big outliers.
- (3) Compute an epoch-difference solution as a reference for the cycle slip detection.
- (4) Check all the observations and find the time intervals which are corrupted by cycle slips.
- (5) If possible, repair the cycle slips.
- (6) Check for clock events.

The phase measurement of the GPS receiver yields a value between 0 and 1 cycle (0 and  $2\pi$ ). After turning on the receiver an integer counter is initialized. During tracking, the counter is incremented by one whenever the fractional phase changes from  $2\pi$  to 0. Thus, for every epoch the accumulated phase is the sum of the directly measured fractional phase and the



integer count. The initial integer number of cycles between the GPS satellite  $i$  and receiver  $k$  is unknown and has to be estimated (see observation equations). In case of

$$n_{Fk}^i(t_2) - n_{Fk}^i(t_1) \neq 0, \quad (8.2-170)$$

a cycle slip has occurred between time  $t_1$  and  $t_2$ . MAUPRP uses a cycle slip detection and correction algorithm which is as followed:

If we define the epoch-difference of the  $L_1$  residual as  $r_1$  and that for  $L_2$  as  $r_2$ , it is written for two different GPS satellites as

$$\begin{aligned} r_1 &= b_1 \lambda_1 + (I_k^{ij}(t_2) - I_k^{ij}(t_1)) + (\delta_k(t_2) - \delta_k(t_1)) \\ r_2 &= b_2 \lambda_2 + \frac{f_1^2}{f_2^2} (I_k^{ij}(t_2) - I_k^{ij}(t_1)) + (\delta_k(t_2) - \delta_k(t_1)), \end{aligned} \quad (8.2-180)$$

where the values  $b_1$  and  $b_2$  are integers used to test whether a no-cycle slip hypothesis ( $b_1 = b_2 = 0$ ) holds. The residual in  $L_3$  linear combination is computed as

$$r_3 = \frac{f_1^2}{f_1^2 - f_2^2} r_1 - \frac{f_2^2}{f_1^2 - f_2^2} r_2 = \beta_1 r_1 + \beta_2 r_2. \quad (8.2-190)$$

One of the two conditions in the no-cycle slip conditions are then

$$|r_3| \leq 3g \sqrt{(\beta_1 \sigma_1)^2 + (\beta_2 \sigma_2)^2}, \quad (8.2-200)$$

where the factor  $g$  is  $\sqrt{2^2}$  caused by the growth in standard deviation by a factor  $\sqrt{2}$  by forming every difference (two satellites and two epochs).  $\sigma_1$  and  $\sigma_2$  is the a priori RMS error of the zero difference observable. The other no-cycle slip condition is linked to the ionospheric refraction behavior. If the mean value  $m$  fulfills

$$m = \frac{1}{2} \left( r_1 + \frac{f_2^2}{f_1^2} r_2 \right) \leq M_{ion} \quad (8.2-210)$$

where  $M_{ion}$  is the maximum ionospheric change from epoch to epoch (specified by the user of MAUPRP), and the condition in Eqn. (8.2-210) hold, then the no-cycle slip hypothesis is accepted as true.

In the opposite case a search over the values  $b_1$  and  $b_2$  is performed. All combinations of

$$\begin{aligned} b_{1p} &= INT\left(\frac{r_1}{\lambda_1}\right) + p, & p &= -J_1, \dots, -1, 0, 1, \dots, J_1 \\ b_{2q} &= INT\left(\frac{r_1}{\lambda_1} - \frac{r_2}{\lambda_2}\right) + q, & q &= -J_2, \dots, -1, 0, 1, \dots, J_2 \end{aligned} \quad (8.2-220)$$

are formed.  $INT$  is the nearest integer and the search ranges  $J_1, J_2$  has to be specified by the user. If one combination meets the no-cycle slip hypothesis, this pair is assumed to be the cycle slip correction. If no combination is found a new ambiguity parameter should be introduced in the parameter estimation program GPSEST. Too many ambiguity parameters (and few GPS satellite observation per epoch) would result in large RMS errors in the other parameter estimated.

In the non-parameters screening of phase observation high-rate precise GPS clocks correction are essential needed. Also due to the rapidly changing GPS satellite geometry as seen by the LEO several time interval criterions have to be reduced. The screening may be iterated when an improved a priori orbit gets available in the final pre-processing.

### GPSEST

GPSEST is the main estimation program for the adjustment of the model parameters. It sets up the linearized observation equations in the Gauss-Markoff Model of full rank and solves the normal equations using Least-Squares Estimation. Since the observation equations are linearized by a Taylor series expansion around 0, the use of good a priori positions are essential to adapt a globally extremum and not a local.

Table 8-1 summarizes the adjustable parameters implemented in GPSEST relevant for processing a LEO orbit. The parameterization as a function of time has three different forms: A) A parameters may be constant over the entire computed arc. This may be the case for ex. antenna offset or patterns. B) A parameter may be represented as a piece-wise linear function in time. This is the case for e.g. the ionosphere parameters and the radiation pressure parameters. C) A parameters may be valid for a single observation epoch only. This is the case for most of the main relevant parameters for a LEO determination, e.g. coordinates, ambiguity resolution after a cycle slip, the stochastic orbit parameters etc.

The constant parameter are either set by the user or estimated once at the beginning of the arc. The piece-wise linear parameter functions in the time interval between  $t_1$  and  $t_2$  are parameterized either by the values in time  $t_1$  and  $t_2$  (two biases) or by the function in  $t_1$  and its time derivative (bias and tilt terms) at equidistant nodal points. The parameter spacing can be defined by the user. Epoch parameters are set up for each epoch. Due to their large number it is generally necessary to pre-eliminate them epoch wise e.g. in CODSP or MAUPRP. This is possible because epoch-parameters, by definition, are not directly correlated – physical correlations are neglected. Due to the large number of coordinate parameters including other epoch parameter, a back-substitution step is necessary to resolve the parameters with the smallest RMS possible. This is done by solving the normal equations for the parameters, but not for the coordinates, and insert them back into the observation equation which then are successively solved for each epoch.

In the use of GPSEST in the pre-processing the main relevant estimated parameters are the epoch specific coordinates, the phase ambiguities and the stochastic ionospheric parameters. If, however, some of the pre-estimated parameters not are well estimated, these may also go into the parameter estimation. But the use of the pseudo-stochastic parameters shall not be introduced in this step. To keep the generation of the a priori orbit as dynamical as possible, the stochastic pulses are first introduced in the reduced-dynamic processing step. 15 min pulses may, however, been used if the RMS of the a priori orbit is worse than some meters.

Parameter	Description
Epoch specific coordinates	Estimation of the position of the LEO assigned to each epoch.
Phase Ambiguities	One initial phase ambiguity parameter has to be assigned for each continued data-locked satellite. Also if a cycle slip occurs that MAUPRP could not repair a new ambiguity parameter is introduced. Resolved ambiguities may be introduced in subsequent program runs.
Stochastic ionospheric parameters	Epoch- and satellite-specific ionosphere parameters may be introduced, together with a priori weights, to support ambiguity resolution.
Radiation pressure parameters	A total of nine parameters per arc may be introduced, three in each of the three orthogonal directions. Constant and periodic once-per-revolution terms may be introduced.
Pseudo-Stochastic Orbit Parameters	Velocity changes in pre-determined directions at user defined epochs may be introduced. Up to three directions, radial, along-track and out-of-plane, may be defined per epoch.
Receiver clock corrections	Estimation using code and phase information if for some reason the CODSP estimates are not good or not estimated.
Receiver antenna phase center variations	Antenna phase center variations may be modeled using different techniques. Model parameters may be determined.

Table 8-1 Parameter types implemented in GPSEST, relevant for LEO orbit determination.

### ORBGEN

In the last step of the pre-processing ORBGEN reads the orbit parameters element file written by GPSEST and updates the standard orbit by numerical integration. It is essential that the orbit models used for the generation of the a priori orbit and for updating of the orbit are identical.

In order to an additional a priori orbit improvement, the last three steps in the pre-processing (MAUPRP -> GPSEST -> ORBGEN) have to be iterated 2-3 times. Some of the constraints and rejection criterions in MAUPRP and GPSEST may slightly be sharpened in order to archive the smallest RMS value on the a priori orbit. The importance of a very reliable a priori orbit, an effective data editing, and orbit adjustment was also a main conclusion in formatting the MOD on CHAMP. This is exactly the task of this iterative step-wise approach in the pre-processing by Bernese.

#### **8.2.4.4 The Reduced-Dynamic Processing**

The final step in the orbit determination of the LEO satellite is the reduced-dynamic processing step. Due to the dynamical model implementation in ORBGEN (which is not optimized for LEO's) it is generally preferable to estimate the orbit using the reduced-dynamic approach and not a kinematic approach. However, a kinematic solution is possible using GPSEST for a kinematic point positioning and hereafter the KINPRE to convert the output into a precise orbit file, but some dynamics will be introduced since the a priori orbit

(which also is essential for a good kinematic approach) is generated by the use of dynamic parameters.

Essential the reduced-dynamic orbit determination is an orbit improvement process where orbital parameters are refined using observations. This is done by ORBGEN and GPSEST and involves three steps

1) **Preparation of a priori orbit information:**

A priori orbit as well as the partial derivatives of the orbit positions with respect to the parameters to be estimated are generated with program ORBGEN by numerical integration of the equations of motion and of the variational equations. The information is written to a standard orbit and a radiation pressure file.

2) **Estimation of improvements for orbit parameters:**

Program GPSEST reads the a priori orbit and the derivatives from these two files, builds up the normal equations in a loop over all observations, and solves the equation system to obtain the parameter improvements. The improved orbital parameters are written to an element file referring to the start epoch of the a priori orbit (osculation epoch).

3) **Update of orbit:**

ORBGEN reads the element file written by GPSEST and integrates the orbit numerically based on the improved orbit parameters. The output is a standard orbit for a specified time interval.

The input data from to the reduced-dynamic processing is the RINEX observation files, the ERP data, Bernese internal auxiliary data, the processed GPS orbits from the Orbit Pert, and the a priori Leo orbit from the Pre Processing part.

ORBGEN

The same procedure has to be followed as in the pre processing. Not only the equations of motion but also the so-called variational equations, which are the derivatives of satellite positions with respect to orbit parameters, have to be integrated numerically for orbit improvement. The integration process writes the polynomial coefficients for the satellite, each component, and each integration subinterval into the standard orbit file and those for all the partials into a radiation pressure file.

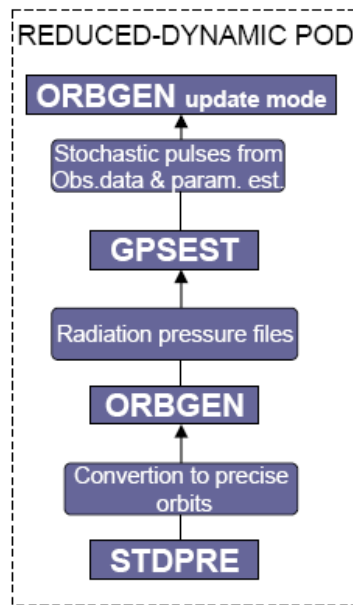


Figure 8-6 The Reduced-Dynamic approach in the orbit determination.

### GPSEST

The actual orbit improvement has to be set up in program GPSEST. It is recommended using data spans shorter than one day. The main difference in this run of GPSEST compared with that of the pre processing is the use of the pseudo-stochastic orbit parameters pulses. They have to be set up for all three directions: radial, along-track, and out-of-plane. It is recommended for LEO POD to use around 6 min pulses in all three directions. If, however, the RMS in some of the directions, radial for instance, is remarkable worse, higher pulses frequencies may be added additional in the radial direction. It is important that the stochastic pulses epoch have to coincide with the integration interval boundary in ORBGEN.

### 8.3 Accelerometer – ACC

#### 8.3.1 Introduction

The Level 1b instrument processing of the ACC is described in [AD13] and briefly repeated here. In case of discrepancies, this document shall take precedence.

#### 8.3.2 Input and Output Parameters

Inputs					
Element		Description	Unit	Range	
L e v e l  1 a	A C C	<b>a<sub>lin</sub></b>	Acceleration in translation	eu	0 – 2 <sup>24</sup> -1
		<b>a<sub>ang</sub></b>	Acceleration in rotation	eu	0 – 2 <sup>24</sup> -1
		<b>p<sub>lin</sub></b>	Translational position of proof mass	m	±10 <sup>-5</sup>
		<b>p<sub>ang</sub></b>	Rotational position of proof mass	rad	±6·10 <sup>-4</sup>
		<b>T</b>	Temperatures, vector of length 6	°C	-50 – 100
		V <sub>pLTC1043</sub>	LTC1043 positive voltage	V	0 – 6
		V <sub>nLTC1043</sub>	LTC1043 negative voltage	V	-6 – 0
		U <sub>pol</sub>	Polarization voltage	V	0 – 12
		Mode	Instrument mode		1 – 7
		ALSwitch	Angular/Linear switch		0, 1
		ADNR	Acceleration data not ready flags, 3x2 matrix		0, 1
	HDNR	Housekeeping data not ready flag		0, 1	
	A O C S	Prop_HK	Cold gas system pressures	Bar	-30 – 60
			Cold gas system temperatures	°C	
Accumulated thruster on-time			s		
Mag_HK		Magneto torquer currents telemetry	A	±20	
Thru_HK	Thruster firing telemetry				
Bus.State		Bus states – for heater switching and PDCU power states			
C C D B	S t r u c t u r e	<b>ACC_r</b>	Position of ACC sensor in S/C frame	m	±5
		<b>ACC_q</b>	Transformation of S/C to ACC sensor frames		±1
		SC_CoG	Parameters to calculate CoG of S/C		
		SC_Cross Section	Structure for estimating effective cross sections, 2-D look-up table as function of azimuth and elevation, providing outward normal vectors in S/C frame		

<b>Inputs</b>					
<b>Element</b>		<b>Description</b>	<b>Unit</b>	<b>Range</b>	
	SC_Mass	Parameters to calculate actual mass of S/C			
	SC_RadP	Structure for estimating effective radiation properties, 2-D look-up table as function of azimuth and elevation, S/C frame			
	SC_Uplift	Look-up table (1-D) for estimate of atmospheric up-drift depending on altitude			
A O C S	ThrusterForce	Nominal thruster forces Attitude control, primary, 1-8 (ACT_An) Orbit control, primary, 1-4 (OCT_An) Attitude control, secondary, 1-8 (ACT_Bn) Orbit control, secondary, 1-4 (OCT_Bn)	mN	±50	
	A C C	Daa	ACC delay between angular acceleration measurement time and time stamp, vector	s	±1
		Dla	ACC delay between linear acceleration measurement time and time stamp, vector	s	±1
		t_C	Duration of ADC conversion	s	±1
t_AD		Delay of ADC analogue part and digital filter	s	±5	
	Aao	Bias of angular acceleration	rad/s <sup>2</sup>	±10 <sup>-2</sup>	
	Aaot	Angular acceleration bias as function of temperatures, vector	rad/s <sup>2</sup> °C	±10 <sup>-4</sup>	
	Aao_tab	Look-up table (LUT1d) of a priori biases of angular acceleration as function of time	rad/s <sup>2</sup>	±10 <sup>-3</sup>	
	Aas	Scale of angular acceleration	rad/s <sup>2</sup> eu	±10 <sup>-9</sup>	
	Aast	Angular acceleration scale as function of temperatures, vector	rad/s <sup>2</sup> eu°C	±10 <sup>-11</sup>	
	Aas_tab	Look-up table (LUT1d) of a priori scales of angular acceleration as function of time	rad/s <sup>2</sup> eu	±10 <sup>-10</sup>	
	Lao	Bias of linear acceleration	m/s <sup>2</sup>	±10 <sup>-4</sup>	
	Laot	Linear acceleration bias as function of temperatures, vector	m/s <sup>2</sup> °C	±10 <sup>-6</sup>	
	Lao_tab	Look-up table (LUT1d) of a priori biases of linear acceleration as function of time	m/s <sup>2</sup>	±10 <sup>-5</sup>	
	Las	Scale of linear acceleration	m/s <sup>2</sup> eu	±10 <sup>-11</sup>	
	Last	Linear acceleration scale as function of temperatures, vector	m/s <sup>2</sup> eu°C	±10 <sup>-13</sup>	
	Las_tab	Look-up table (LUT1d) of a priori scales of	m/s <sup>2</sup> eu	±10 <sup>-12</sup>	

Inputs					
Element		Description	Unit	Range	
		linear acceleration as function of time			
	T <sub>0</sub>	“Standard” temperature of ACC instrument	°C	0 – 30	
	N_Invalid	Numbers of neighbouring measurements to be flagged in cases of outliers		0 – 10	
	N_Invalid <sub>xxx</sub>	Numbers of neighbouring or subsequent measurements to be removed in cases of S/C events invalidating ACC measurements		0 – 100	
	Limits	<i>Structure holding limits for outlier detection and/or rejection</i>			
		Aa_lim	Absolute limit of angular acceleration	eu	
		La_lim	Absolute limit of linear acceleration	eu	
		<b>T_Range</b>	Range of temperatures	°C	
		n <sub>sigma</sub>	σ-factor for detecting outliers		1 – 10
		N <sub>window</sub>	Number of samples in outlier detection fits		1 – 60
		degree	Polynomial degree in outlier detection fits		1 – 8
	Thermistor_Weights	wT12	Weight of temperature 1 wrt temperatur 2		0 – 1
		wT34	Weight of temperature 3 wrt temperatur 4		0 – 1
		wT56	Weight of temperature 5 wrt temperatur 6		0 – 1
	Apply Correction	Radiation Pressure	Flag indicating if solar radiation pressure estimate is subtracted from Level 1b data		0, 1
		Gravity Gradient	Flag indicating if gravity gradient estimate is subtracted from Level 1b data		0, 1
		Centrifugal	Flag indicating if centrifugal acceleration estimate is subtracted from Level 1b data		0, 1
	Constants	c	Speed of light (=299,792,458 m/s)	m/s	
		GM <sub>E</sub>	Earth’s gravitational constant (=3.986004418 · 10 <sup>14</sup> m <sup>3</sup> /s <sup>2</sup> )	m <sup>3</sup> /s <sup>2</sup>	
		P <sub>Sun</sub>	Mean radiation pressure of solar flux (~1366 W/m <sup>2</sup> )	W/m <sup>2</sup>	
r <sub>E</sub>		Mean radius of Earth (~6.372· 10 <sup>6</sup> m)	m		
<b>r</b>		Position of CoG, ITRF, at ACC time instants (cubic interpolation of Level 1b.Eph.r)	m	±10 <sup>7</sup>	
<b>v</b>		Velocity of CoG, ITRF, at ACC time instants (cubic interpolation of Level 1b.Eph.v)	m/s	±10 <sup>4</sup>	



Inputs			
Element	Description	Unit	Range
$\mathbf{q}_{\text{ITRF} \leftarrow \text{S/C}}$	Attitude of S/C, S/C $\rightarrow$ ITRF transformation at ACC time instants		$\pm 1$
$\dot{\mathbf{q}}_{\text{ITRF} \leftarrow \text{S/C}}$	Time derivative of $\mathbf{q}_{\text{ITRF} \leftarrow \text{S/C}}$	s <sup>-1</sup>	$\pm 1$

Outputs					
Element		Description	Unit	Range	
L e v e l 1 b	A C C	t	Time, UTC	s	
		<b>a</b>	Pre-processed linear acceleration data <sup>17</sup>	m/s <sup>2</sup>	$\pm 10^{-4}$
		<b>a</b> <sub>ang</sub>	Pre-processed angular acceleration data	rad/s <sup>2</sup>	$\pm 10^{-2}$
		<b>p</b>	Position of proof mass within ACC cavity	m	$\pm 10^{-5}$
		<b>p</b> <sub>ang</sub>	Angular position of proof mass within ACC cavity	rad	$\pm 10^{-3}$
		<b>Temp</b>	Vector of 6 temperatures of the ACC	°C	-50 – 100
		V <sub>pLTC1043</sub>	LTC1043 positive voltage	V	0 – 6
		V <sub>nLTC1043</sub>	LTC1043 negative voltage	V	-6 – 0
		U <sub>pol</sub>	Polarization voltage	V	0 – 12
		<b>a</b> <sub>centr</sub>	Centrifugal acceleration	m/s <sup>2</sup>	$\pm 10^{-7}$
		<b>a</b> <sub>GG</sub>	Gravity gradient acceleration	m/s <sup>2</sup>	$\pm 10^{-7}$
		Flags <sub>ACC</sub>	Flags related to the ACC measurement		0 – 2 <sup>18</sup> -1
	Flags <sub>Platform</sub>	Flags related to the S/C platform		0 – 63	
	S C D Y N	<b>a</b> <sub>Sun</sub>	Acceleration due to Solar radiation pressure	m/s <sup>2</sup>	$\pm 10^{-7}$
		<b>e</b> <sub>Sun</sub>	Direction to the Sun, unit vector, S/C frame		$\pm 1$
		<b>A</b> <sub>Xxx</sub>	Cross section normal vectors of the S/C, Xxx = head, down, left, right	m <sup>2</sup>	0 – 10
		<b>K</b> <sub>Earth</sub>	Normalized downwards reflectivity	m <sup>2</sup>	$\pm 0.1$
		m <sub>S/C</sub>	Mass of S/C	kg	300 – 500
		<b>r</b> <sub>CoG</sub>	Center of gravity, S/C frame	m	$\pm 2$
<b>P</b> <sub>Gas</sub>	Pressure of gas tanks	Pa	0 – 10 <sup>7</sup>		

<sup>17</sup> All accelerations in Level 1b.ACC are provided in the S/C frame.

Outputs				
Element		Description	Unit	Range
	$T_{\text{Gas}}$	Temperature of gas tanks	°C	-50 – 70
	Flags <sub>Platform</sub>	Flags related to the S/C platform (indicates if all inputs were available or one or more was missing)		0 – 63
	dt_thr	Thruster on-time in seconds (Start of on-time at “Timestamp”), field with 12 columns (column 1 = ACT 1, ..., column 9 = OCT 1, ...)	s	0 – 10 <sup>5</sup>
	f_thr	Nominal thrust force of activated thrusters (combined force), field with 3 columns	mN	-240 – 170
	thr_set	Flag indicating which thruster branch was active (= 0 for no thrusters powered, = 1 for main units powered, = 2 for redundant units powered, =3 for both main and redundant units powered)		0 – 3

### 8.3.3 ACC Algorithms

#### 8.3.3.1 Overview

The intrinsic ACC processing is described in [AD13], cf. Appendix L. The Level 1a data comprising the acceleration vectors (linear and rotational) in engineering units (eu) and the required house-keeping information (e.g. temperatures and polarization voltage) in physical units. The Level 1a data shall (optionally, i.e. user selectable) be stored as Level 1a Products.

The processing of Level 1a data to Level 1b ACC data is sketched in Figure 8-7 below.

It consists of the following tasks:

- Correct for delays
- Identification of spurious samples, flag measurements accordingly
- Determine satellite events with possible impact on ACC measurements (e.g. thruster firing, magnetic torquer operation, heater switching) and flag ACC measurements accordingly
- Conversion of ACC measurements (and their linear estimates) from engineering units (eu) to near physical units
- Rotation of accelerations from ACC frame to S/C frame - disabled
- Calculate and possibly (depending on processor configuration) apply correctional terms – application disabled

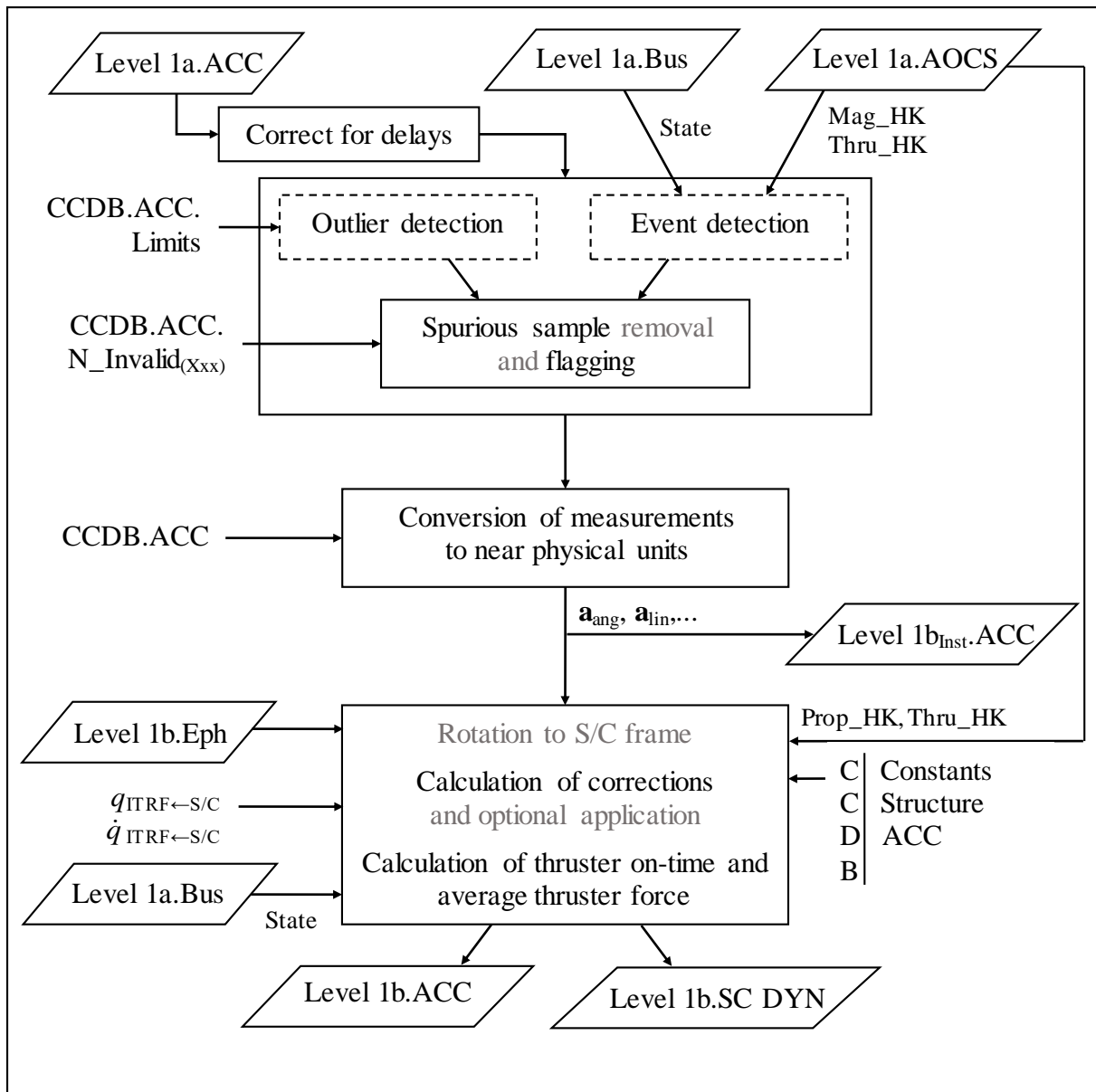


Figure 8-7 ACC Processing Overview

### 8.3.3.2 Outlier and Event Detection

The ACC outlier detection is described in [AD13]. This method is to be applied to the accelerations and temperatures. Initially these quantities are checked to be within the corresponding CCDB.ACC.Limits values – the acceleration components shall be numerically less than the corresponding Aa\_Lim and La\_Lim values. Then, the (remaining) measurements are checked not to deviate significantly from linear prediction of the adjacent samples. The principle is to make N+1 linear fits of the N adjacent samples (excluding the target sample itself) and if  $n \cdot \sigma_k \geq |x - x_{est,k}|$  for all k – for all N+1 linear fits – then the target sample is characterized as spurious. In this:

$$N = \text{CCDB.ACC.Limits.N}_{\text{window}}$$

$$n = \text{CCDB.ACC.Limits.n}_{\text{sigma}}$$

$$k = \text{index of linear fit, } k = 0..N$$

$\sigma_k$  = sigma of linear fit k  
 x = target sample (measurement)  
 $x_{est,k}$  = estimate of linear fit k at target sample

In case a sample is detected as an outlier, additionally CCDB.ACC.N\_invalid samples before and after are flagged

S/C events causing distortion of the ACC measurements are detected and the affected ACC samples plus a number of subsequent samples are flagged. The list of events is given in Table 8.3-1 below. The column “**Xxx**” is the CCDB.ACC.N\_Invalid suffix, where N\_Invalid<sub>xxx</sub> is the number of subsequent samples to flag. Thruster activation is computed using the State<sub>Thruster</sub> function described in Section 6.1.1, Eq. (6-1.16) using zero for the st (settling time) parameter.

Description	Level 1a Source	Xxx	Comment
Thruster activation	AOCS.Thru_HK	Thru	After activation has ceased
Heater switching	Bus.State	Heat	

Table 8.3-1 S/C Events Affecting ACC Measurements

### 8.3.3.3 Conversion of Measurements to Physical Units

The calibration parameters of the ACC are applied to convert the acceleration measurements in engineering units to physical units. Both raw measurements as well as estimated values of the linear regressions are to be converted. The conversion is described in Section 6.3 of [AD13] with the modification that the temperatures are not weighted equally but using CCDB.Acc.Thermistor\_Weights.wT12, wT34, and wT56 respectively such that

$$T_X = wT12 T_1 + (1-wT12) T_2 \tag{8.3-4}$$

$$T_Y = wT34 T_3 + (1-wT34) T_4 \tag{8.3-6}$$

$$T_Z = wT56 T_5 + (1-wT56) T_6 \tag{8.3-8}$$

$T_X$ ,  $T_Y$ , and  $T_Z$  are multiplied onto the respective components of CCDB.ACC.**Laot** and CCDB.ACC.**Last**. However, for the angular acceleration parameters,  $T_Y$  is multiplied onto the x-components of CCDB.ACC.**Aaot** and CCDB.ACC.**Aast**,  $T_Z$  onto the y-components, and  $T_X$  onto the z-components.

With respect to delays contained in CCDB.ACC.**Daa** (angular) and CCDB.ACC.**Dla** (linear), only the delay of the linear x-component is used, i.e.  $Dla_1$ . All measurements (including temperatures and voltages) are corrected using this delay, i.e. the common time-stamp is:

$$t_{out,ACC} = t_{0,ACC} + CCDB.ACC.t_C/2 - CCDB.ACC.t_{AD} - CCDB.ACC.Dla_1 \tag{8.3-10}$$

No interpolations (e.g. to account for differences in the delays) are performed.

### 8.3.3.4 Mass of SC

See E.3.

### 8.3.3.5 Center of gravity

See E.3.

### 8.3.3.6 Rotation to S/C Frame

The linear and angular accelerations (measured and estimated values) are rotated to the S/C frame by means of the `CCDB.Structure.ACC_q`:

$$\mathbf{a}_{\text{lin},S/C} = \text{CCDB.Structure.ACC\_q}^{-1} \times \mathbf{a}_{\text{lin},\text{ACC}} \quad (8.3-12)$$

$$\mathbf{a}_{\text{ang},S/C} = \text{CCDB.Structure.ACC\_q}^{-1} \times \mathbf{a}_{\text{ang},\text{ACC}} \quad (8.3-15)$$

where  $\mathbf{a}_{\dots,\text{ACC}}$  are the accelerations in the ACC frame (from Level 1b<sub>Inst</sub>) and  $\mathbf{a}_{\dots,S/C}$  are the accelerations in the S/C frame (going to Level 1b).

### 8.3.3.7 Corrections and Cross Sections

The correctional terms of the accelerations are described next. Whether these are applied to the Level 1b acceleration data or not depends on the values of the `CCDB.ACC.ApplyCorrection` elements (see Eq. (8.3-120)). The actual state of these are noted in the `FlagsACC`.

The corrections are as follows:

- Calculate bias due to centrifugal acceleration, for background information see [AD13].
  - Calculate deviation of ACC proof mass position from center of gravity

$$\mathbf{r}' = \text{CCDB.Structure.ACC\_r} - \mathbf{r}_{\text{CoG}} \quad (8.3-30)$$

- Calculate the angular velocity relative to Earth

$$\vec{\omega}_{S/C,ITRF} = \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} = \begin{pmatrix} 2(\dot{q}_1 q_4 - \dot{q}_2 q_3 + \dot{q}_3 q_2 - \dot{q}_4 q_1) \\ 2(\dot{q}_2 q_4 - \dot{q}_3 q_1 + \dot{q}_1 q_3 - \dot{q}_4 q_2) \\ 2(\dot{q}_3 q_4 - \dot{q}_1 q_2 + \dot{q}_2 q_1 - \dot{q}_4 q_3) \end{pmatrix} \quad (8.3-40)$$

where  $\mathbf{q}$  and  $\dot{\mathbf{q}}$  are  $\mathbf{q}_{\text{ITRF} \leftarrow S/C}$  and  $\dot{\mathbf{q}}_{\text{ITRF} \leftarrow S/C}$  respectively.

- Correct for angular velocity of Earth:

$$\vec{\omega} = \vec{\omega}_{S/C,ITRF} - \mathbf{q}_{\text{ITRF} \leftarrow S/C}^{-1} \times (0 \ 0 \ 2\pi/86400)^T \quad (8.3-45)$$

- Calculate centrifugal acceleration

$$\mathbf{a}_{\text{centr}} = -\vec{\omega} \times (\vec{\omega} \times \mathbf{r}') \quad (8.3-50)$$

- Calculate bias due to gravity gradient. For this, transformation via the local NEC frame is needed.  $\mathbf{a}_{\text{GG}}$  in S/C frame is computed follows, where  $\mathbf{r}'$  is the deviation of ACC proof mass position from center of gravity, cf. Eq. (8.3-30),  $\mathbf{r}$  is the position of center of gravity in ITRF, and  $\mathbf{q}_{\text{NEC} \leftarrow \text{ITRF}}$  is the quaternion representation of the  $\mathbf{R}_{\text{NEC} \leftarrow \text{ITRF}}$  ITRF  $\rightarrow$  NEC( $\mathbf{r}$ ) transformation matrix, cf. Appendix C:

$$\mathbf{q}_{\text{NEC} \leftarrow S/C} = \mathbf{q}_{\text{ITRF} \leftarrow S/C} \otimes \mathbf{q}_{\text{NEC} \leftarrow \text{ITRF}} \quad (8.3-56)$$

$$\mathbf{r}'_{\text{NEC}} = \mathbf{q}_{\text{NEC} \leftarrow S/C} \times \mathbf{r}' \quad (8.3-58)$$

$$\mathbf{a}_{GG} = \frac{-CCDB.Constants.GM_E}{|\mathbf{r}|^3} \mathbf{q}_{NEC \leftarrow S/C}^{-1} \times \begin{pmatrix} \mathbf{r}'_{NEC,x} \\ \mathbf{r}'_{NEC,y} \\ -2\mathbf{r}'_{NEC,z} \end{pmatrix} \quad (8.3-60)$$

- Calculate acceleration vector due to radiation pressure of solar flux.
  - Determine if the Sun is visible. Let  $\mathbf{r}_{Sun} = \mathbf{x}l_{sun}(t)$  (see [RD12]), then if  $\mathbf{r} \cdot \mathbf{r}_{Sun} < 0$  and  $|\mathbf{r} \times \mathbf{r}_{Sun}|/|\mathbf{r}_{Sun}| < CCDB.Constants.r_E$  the S/C is in the shadow of the Earth, and  $\mathbf{a}_{Sun} = \mathbf{0}$ . Otherwise proceed with the following steps.
  - Determine the line of sight of the Sun in the S/C frame: let  $\mathbf{s}$  be the position of the Sun in the S/C frame (then  $\mathbf{e}_{Sun} = \mathbf{s}/|\mathbf{s}|$ , which is also stored in Level 1b.ACC):

$$\mathbf{s} = \mathbf{q}_{ITRF \leftarrow S/C}^{-1} \times (\mathbf{r}_{Sun} - \mathbf{r}) \quad (8.3-65)$$

- Determine the optical reflectivity properties for direction of incidence of solar radiation,  $\theta$  and  $\varphi$  are obtained by converting  $\mathbf{s}$  to spherical coordinates:

$$\mathbf{K} = \text{Interpol}_{in}(CCDB.Structure.SC\_RadP, \theta, \varphi) \quad (8.3-70)$$

- Calculate the radiation pressure

$$\mathbf{FT} = CCDB.Constants.P_{Sun} \mathbf{K} / CCDB.Constants.c \quad (8.3-80)$$

- Calculate the mass of the spacecraft,  $Mass$ , (also stored in Level 1b.ACC): see Appendix E.

- Calculate acceleration vector due to radiation pressure

$$\mathbf{a}_{Sun} = \frac{\mathbf{FT}}{Mass} \quad (8.3-100)$$

- Calculate the downward reflectivity of the S/C:

$$\mathbf{K}_{Earth} = \text{Interpol}_{in}(CCDB.Structure.SC\_RadP, \theta_E, \varphi_E) \quad (8.3-102)$$

where  $\theta_E$  and  $\varphi_E$  are the spherical coordinates of  $\mathbf{r}_E = -\mathbf{q}_{ITRF \leftarrow S/C}^{-1} \times \mathbf{r}$ .

The corrections calculated above are used to generate the Level 1b accelerations from the instrument Level 1b data,  $\mathbf{a}$  and  $\mathbf{a}_{ang}$ :

$$\text{Level 1b.ACC.}\mathbf{a} = \mathbf{a} - A_{centr} \mathbf{a}_{centr} - A_{Sun} \mathbf{a}_{Sun} - A_{GG} \mathbf{a}_{GG} \quad (8.3-120)$$

$$\text{Level 1b.ACC.}\mathbf{a}_{ang} = \mathbf{a}_{ang} \quad (8.3-130)$$

where  $A_{xxx}$  are the respective  $CCDB.ACC.ApplyCorrection$  flags (zero or one).

The cross sections of the S/C are provided as outward (from S/C) pointing normal vectors with lengths corresponding to projected surface areas. The vectors are calculated by linear interpolations in the  $CCDB.Structure.SC\_CrossSection$  table. The interpolations are performed for the angles ( $\theta$  and  $\varphi$ ) of the spherical representation of the following vectors (orbit frame):

$$\mathbf{Head} = \mathbf{q}_{ITRF \leftarrow S/C}^{-1} \times \mathbf{v}$$

$$\mathbf{Left} = \mathbf{q}_{ITRF \leftarrow S/C}^{-1} \times -(\mathbf{v} \times \mathbf{r})$$

$$\mathbf{Right} = \mathbf{q}_{ITRF \leftarrow S/C}^{-1} \times (\mathbf{v} \times \mathbf{r})$$

$$\mathbf{Down} = \mathbf{q}_{ITRF \leftarrow S/C}^{-1} \times (\mathbf{v} \times (\mathbf{v} \times \mathbf{r}))$$

I.e.

$$\text{Level 1b.ACC.A}_w = \text{Interpol}_{\text{lin}}(\text{SC\_CrossSection}, \theta_w, \varphi_w) \quad (8.3-140)$$

where  $w = \text{Head, Left, Right, Down}$ , and  $\text{Interpol}_{\text{lin}}$  is given in Appendix E.

### 8.3.3.8 Thruster on-time and average thrust per epoch

- Nominal thrust force for thruster branch A (main unit) in mN (Column 1 = ACT1 ... Column 9 = OCT1 ...)

```
f1 = [ 0 0 0 0 10.0 -10.0 10.0 -10.0 -50 -50 0 0
       0 0 0 0 17.3 -17.3 -17.3 17.3 0 0 -50 -50
       -20 20 -20 20 0.0 0.0 0.0 0.0 0 0 0 0]';
```

- Nominal thrust force for thruster branch B (redundant unit) in mN (Column 1 = ACT1 ... Column 9 = OCT1 ...)

```
f2 = [ 0 0 0 0 -10.0 10.0 -10.0 10.0 -50 -50 0 0
       0 0 0 0 -17.3 17.3 17.3 -17.3 0 0 50 50
       20 -20 20 -20 0.0 0.0 0.0 0.0 0 0 0 0]';
```

- Set epoch of SC\_DYN to coincide with the epochs of the onboard AOCS cycle.

```
t_SC_DYN = convert_to_UTC(AOCS.THRU.TCj)
```

- Resample BUS HK data (PDCU state 16 and 52 for thruster branch A and B, respectively) to the epochs of SC\_DYN data using linear interpolation.

```
branch_A_power_state = interp(t_BUS, BUS.STATE(16), t_SC_DYN, 'linear')
branch_B_power_state = interp(t_BUS, BUS.STATE(52), t_SC_DYN, 'linear')
```

- Create flag for thruster branches power state

```
thr_set = branch_A_power_state + 2 * branch_B_power_state
```

- Initialize thruster on-time for current epoch

```
current_dt = zeros(1,12)
```

- Loop over all epochs except the last epoch (loop starts at least 18 hours in the past since this is the longest thruster on-time allowed by the AOCS)

```
for n = 1:N-1
```

- Copy new on-time for the thrusters that received a new on-time command

```
current_dt(AOCS.THRU.Valid(n,:) == true) = ...
    AOCS.THRU.dt(n,AOCS.THRU.Valid(n,:) == true)
```

- Calculate the time difference between the current and next epoch of the AOCS cycle

```
time_diff = AOCS.THRU.TCj(n+1) - AOCS.THRU.TCj(n)
```

- Set the commanded thruster on-time for this epoch and transfer the remaining on-time to the next epochs

```
dt_thr(n,:) = min(current_dt,time_diff)
current_dt = current_dt - SC_DYN.dt_thr(n,:)
```

- Select the thrust force vector depending on which thruster branch is active

```
if thr_set(n) == 1
    f = f1
elseif thr_set(n) == 2
    f = f2
else
    f = zeros(3,12)
end
```

- Calculate the average thrust force vector per epoch (superscript T means transposition of vector)

```
f_thr(n,:) = [(f(1,:) dt_thrT) (f(2,:) dt_thrT) ...
    (f(3,:) dt_thrT)] / time_diff
```

```
end for
```

### 8.3.3.9 Flags

The flags of the Level 1b.ACC product are as listed below. The values of  $Flags_{ACC}$  and  $Flags_{Platform}$  are obtained by summing the values of all applicable conditions.

Value	Condition
$Flags_{ACC}$	
1	$a_{Sun}$ correction applied
4	$a_{GG}$ correction applied



Value	Condition
8	$\mathbf{a}_{\text{centr}}$ correction applied
16	$U_{\text{Pol}}$ detected as outlier
64	Temperature detected as outlier
128	Proof mass angular position ( $\mathbf{p}_{\text{ang}}$ ) detected as outlier
256	Proof mass linear position ( $\mathbf{p}$ ) detected as outlier
512	Angular acceleration ( $\mathbf{a}_{\text{ang}}$ ) detected as outlier
1024	Linear acceleration ( $\mathbf{a}$ ) detected as outlier
<b>Flags<sub>Platform</sub></b>	
4	Heater switching
8	Level 1a.Bus.State telemetry missing
16	Thruster firing
32	Level 1a.AOCS.Thru_HK telemetry missing

## 8.4 ICRF/ITRF Coordinate Transformation

### 8.4.1 Introduction

The International Celestial Reference Frame (ICRF) is a realization of the International Celestial Reference System through observed extragalactic positions on the sky. The origin of the system coincide with the geocenter of the Earth, the Z-axis points towards the north pole, the X-axis towards the First Point of Aries (Equinox), and the Y-axis complete the right hand system. It is defined at the standard epoch denoted by J2000.0.

The realization of the International Terrestrial Reference System is the International Terrestrial Reference Frame (ITRF). It is formed by a number of terrestrial sites where temporal effects (plate tectonics, tidal effects etc.) are also taken into account. In a Cartesian coordinate system, the centre is in the geocenter, the Z-axis coincides with the rotational axis of the Earth, the X-axis points towards the mean Greenwich meridian, and again the Y-axis completes the right hand system. If geographical coordinates (ellipsoidal latitude, longitude, and height) are required instead of Cartesian, the World Geodetic System 1984 (WGS84) is used, which is an Earth-centered, Earth-fixed Cartesian system with a defined reference ellipsoid.

### 8.4.2 Input and Output Parameters

<b>Inputs</b>			
Element	Description	Unit	Accuracy
$\mathbf{r}_{\text{ITRF}}$	Vector of position in the ITRF system.	m	any
$t_{\text{input}}$	Time of the observation.	GPS	

erp	Earth rotation parameters, e.g. from the IGS. The accuracy of the Ultra-rapid is sufficient enough, but since the Bernese SW need the minimum of the rapid-product, it could be convenient to use the same erp.	mas mas/day	<0.3mas <0.5mas /day
-----	---	----------------	----------------------------

Outputs			
Element	Description	Unit	Accuracy
r_ICRF	Vector of position in the ICRF system.	m	~same as input

### 8.4.3 Algorithm Processing

The transformation between ICRF and ITRF is performed by means of rotations. For an arbitrary vector  $\mathbf{r}$ , the transformation between the two frames is given by

$$\mathbf{r}_{ICRF} = \mathbf{Q} \times \mathbf{R} \times \mathbf{W} \times \mathbf{r}_{ITRF} \tag{8.4-10}$$

where

- $\mathbf{Q}$  is nutation, precession, and frame bias matrix
- $\mathbf{R}$  is global apparent Sidereal time rotation matrix
- $\mathbf{W}$  is polar motion matrix.

The component of the three matrices of the coordinate transformation has been summarized in the International Earth Rotation and Reference System Service (IERS) Conventions 2003 [RD4]. A resolution of the International Astronomical Union (IAU) 2000 recommends that, beginning on 1 January 2003, the IAU 1976 Precession Model and IAU 1980 Theory of Nutation be replaced by the precession-nutation model IAU 2000. Two different models exists: the IAU 2000A model with a element accuracy of 0.2 mas level and a shorter and computational more efficient version IAU 2000B model with accuracies at the 1 mas level. For Swarm the IAU 2000B model will be sufficient and used in the following description.

Fortran77 routines that implement the IAU 2000 transformations are provided on the IERS Conventions web page [RD9]. The main routines, its functions and input output argument used for the ICRF/ITRF transformation is listed in Table 8.4-1. The routines are to a large extent self-contained with detailed comments in the code and uses utility routines from the IAU Standards of Fundamental Astronomy software collection, which can be found at [RD9] together with its documentation of the software.

The polar motion matrix,  $\mathbf{W}$ , is expressed in terms of three rotation matrices, where  $\mathbf{R}_1$ ,  $\mathbf{R}_2$ , and  $\mathbf{R}_3$  denote rotation matrices with positive angle about the axes 1, 2, and 3 of the coordinate frame. It is given by

$$\mathbf{W}(t) = \mathbf{R}_3(-s') \times \mathbf{R}_2(x_p) \times \mathbf{R}_1(y_p), \tag{8.4-20}$$

where  $x_p$  and  $y_p$  being the polar coordinates of the Celestial Intermediate Pole (CIP) in the Terrestrial Reference System (TRS) and  $s'$  being a quantity which provides the position of the Terrestrial Ephemeris Origin (TEO) on the equator of the CIP corresponding to the kinematical definition of the non-rotating origin (NRO) in the ITRS when CIP is moving with respect to the ITRS due to polar motion. The expression of  $s'$  as a function of the coordinates  $x_p$  and  $y_p$  is

$$s' = \frac{1}{2} \int (x_p \dot{y}_p - \dot{x}_p y_p) dt, \quad (8.4-30)$$

where the dot over the polar coordinates indicates the time derivative. The  $s'$  is derived by the subroutine SP2000, using the Terrestrial Time (TT), calculated in Julian Day as input. The correlation between TT in JD and GPS time in JD is given as

$$JD_{TT} = JD_{GPS} + 51.184[\text{sec}]. \quad (8.4-40)$$

The polar motion matrix is computed by the subroutine POM2000 using the  $s'$  and the polar coordinates as input. The polar coordinates are read from the erp-tabular files, defined as external auxiliary input data and listed in the input parameters in Section 8.4.2.

For the Global Apparent Sidereal Time rotation matrix, this is given by

$$\mathbf{R}(t) = \mathbf{R}_3(-\theta), \quad (8.4-50)$$

where  $\theta$  being the Earth Rotation Angle between the Celestial Ephemeris Origin (CEO) and the TEO at date  $t$  on the equator of the CIP – also called for GAST. The GAST is derived by the subroutine GST2000 with input of TT, Universal Time (UT1) in Julian Days, and nutation of longitude. The nutation of longitude is estimated by the subroutine NU2000B, where the precession-nutation model IAU 2000B is taken into account. For computation of the UT1 time the difference between UT1 and UTC,  $\Delta\text{UT1}$ , time is needed. This is also distributed in the erp-tabular file together with the polar coordinates. The relation is

$$JD_{\text{UT1}} = JD_{\text{UTC}} + \Delta\text{UT1}. \quad (8.4-60)$$

Main Subroutine	Function	Inputs	Outputs
SP2000	Calculates the quantity $s'$ .	<i>DATE1, DATE2:</i> - TT (Terrestrial Time) date  NOTE. For Julian Day (JD): $JD = \text{DATE1} + \text{DATE2}$	<i>s_mark:</i> - The quantity $s'$
POM2000	Form polar motion matrix	<i>Xp, Yp:</i> - Coordinates of the pole <i>SP:</i> - The quantity $s'$	<i>W:</i> - The 3x3 polar motion matrix
NU2000B	Compute the nutation angle on longitude and obliquity, according to the IAU2000B theory	<i>DATE1, DATE2:</i> - Same as for SP2000	<i>Dpsi:</i> - Nutation in longitude <i>Depr:</i> - Nutation in obliquity
CBPN2000	Compute the classical bias-precession-nutation matrix	<i>DATE1, DATE2:</i> - Same as for SP2000 <i>Dpsi, Depr:</i> - Nutation in longitude and obliquity	<i>Q:</i> - The 3x3 bias-precession-nutation matrix
GST2000	Compute the Greenwich Apparent	<i>UTA, UTB:</i> - Universal time UT1	<i>GAST:</i> - The Greenwich

Main Subroutine	Function	Inputs	Outputs
	Sidereal Time	date <i>DATE1, DATE2:</i> - Same as for SP2000 <i>Dpsi:</i> - Nutation in longitude  NOTE. Similar to TT: JD = UTA + UTB	Apparent Sidereal Time
T2C2000	Compute the rotation matrix between ITRF and ICRF	<i>W:</i> - The 3x3 polar motion matrix <i>GAST:</i> - The Greenwich Apparent Sidereal Time <i>Q:</i> - The 3x3 bias-precession-nutation matrix	<i>R_T2C:</i> - The 3x3 rotation matrix between ITRF and ICRF

Table 8.4-1 The main routines from the IERS Conventions webpage for the ITRF to ICRF transformation.

The nutation, precession, and frame bias matrix is given as

$$\mathbf{Q}(t) = \begin{pmatrix} 1 - aX^2 & -aXY & X \\ -aXY & 1 - aY^2 & Y \\ -X & -Y & 1 - a(X^2 + Y^2) \end{pmatrix} \times R_3(s), \tag{8.4-70}$$

where  $s$  being a quantity which provides the position of the CEO on the equator of the CIP corresponding to the kinematical definition of the NRO in the Geocentric Celestial Reference System(GCRF) when the CIP is moving with respect to the GCRS, between the reference epoch and the epoch  $t$  due to the precession and nutation.  $X$  and  $Y$  is the coordinates of the CIP in the Celestial Reference System given by

$$X = \sin d \cos E, \quad Y = \sin d \sin E, \quad Z = \cos d, \tag{8.4-80}$$

which also defines the quantities  $d$  and  $E$ .  $a$  of Eq. (8.4-70) is given as

$$a = \frac{1}{(1 + \cos d)}. \tag{8.4-90}$$

The quantities of the elements of the  $\mathbf{Q}$  matrix are calculated in the subroutine CBPN, which uses the TT and nutation in longitude and obliquity. The nutation in longitude and obliquity is estimated in the subroutine of NU2000B, like for the computation of the nutation, precession, and frame bias matrix.

The subroutine T2C2000 computes the rotation matrix between ITRF and ICRF.

## Appendix A

### Abbreviations and Acronyms

ACC	Accelerometer
ADC	Analogue to Digital Converter
AOCS	Attitude and Orbit Control System
ASM	Absolute Scalar Magnetometer
CCDB	Characterisation and Calibration Data Base
CHU	Camera Head Unit
CDC	Compact Detector Coil
CSC	Compact Spherical Coil
CRF	Common Reference Frame of the STR
DNSC	Danish National Space Center
EADS	European Aeronautic Defence and Space
EES	End-to-End System Simulator
EFI	Electric Field Instrument
ESA	European Space Agency
eu	engineering unit
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GPSR	GPS/GNSS Receiver
ICRF	International Celestial Reference Frame
IMDD	Instrument Measurement Data Definition
ISP	Instrument Source Packet
ITRF	International Terrestrial Reference Frame
NEC	North-East-Centre reference frame
N/A	Not Applicable
PPS	Pulse Per Second
S/C	Spacecraft
SRD	System Requirements Document
STR	Star Tracker
TBC	To Be Confirmed
TBD	To Be Defined
UTC	Universal Time Coordinated
VFM	Vector Field Magnetometer

## Appendix B

### Quaternions

A quaternion is a 4 dimensional hyper complex number. In this document, we use normalized quaternions to describe rotations. A rotation  $\theta$  (degrees) around a unit vector  $\mathbf{e}$  is described by the quaternion:

$$\mathbf{q} = (e_x \sin(\theta/2), e_y \sin(\theta/2), e_z \sin(\theta/2), \cos(\theta/2)) \quad (\text{B-10})$$

Example: the  $30^\circ$  rotation around the vector  $\mathbf{e} = (1/\sqrt{2}, 1/\sqrt{2}, 0)$  is depicted in Figure B - 1 as the rotation of the blue frame A into the red frame B –  $\mathbf{z}_B$  is tilted out of the  $\mathbf{y}_A$ - $\mathbf{z}_A$  frame towards the reader. This rotation is given by the quaternion

$$\mathbf{q}_{B \leftarrow A} = (\sin(30^\circ/2)/\sqrt{2}, \sin(30^\circ/2)/\sqrt{2}, 0, \cos(30^\circ/2)) \approx (0.183, 0.183, 0, 0.966).$$

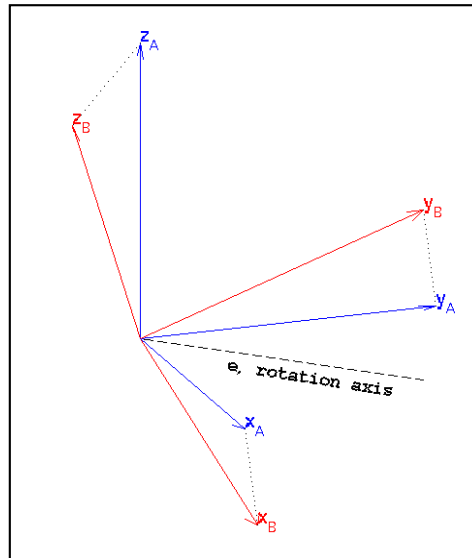


Figure B - 1 A  $30^\circ$  Rotation

To transform the representation of a vector,  $\mathbf{v}$ , from frame A to frame B, calculate Eq. (B-20) and extract the first three components (the fourth component is 0):

$$\mathbf{v}^B = (\mathbf{q}_{B \leftarrow A}^* \otimes (v_x^A, v_y^A, v_z^A, 0) \otimes \mathbf{q}_{B \leftarrow A})_{1..3} \quad (\text{B-20})$$

where

$$\mathbf{q}^* \equiv \mathbf{q}^{-1} = (-q_1, -q_2, -q_3, q_4) \quad (\text{B-30})$$

The multiplication ( $\otimes$ ) of quaternions is most elegantly written using the “complex” form of quaternions:

$$\mathbf{q} = q_1 \mathbf{i} + q_2 \mathbf{j} + q_3 \mathbf{k} + q_4 \quad (\text{B-40})$$

where  $\mathbf{i}$ ,  $\mathbf{j}$ ,  $\mathbf{k}$  are the “complex” unit vectors such that:

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1 \quad (\text{B-50})$$

$$\mathbf{ij} = -\mathbf{ji} = \mathbf{k} \quad (\text{B-60})$$

The full formula for multiplication is given by:

$$\mathbf{p} \otimes \mathbf{q} = \begin{pmatrix} p_1 q_4 + p_2 q_3 - p_3 q_2 + p_4 q_1 \\ -p_1 q_3 + p_2 q_4 + p_3 q_1 + p_4 q_2 \\ p_1 q_2 - p_2 q_1 + p_3 q_4 + p_4 q_3 \\ -p_1 q_1 - p_2 q_2 - p_3 q_3 + p_4 q_4 \end{pmatrix}^T \quad -\mathbf{ji} = \mathbf{k} \quad (\text{B-70})$$

In the example of Figure B - 1 the transformation of the vector (0, 0, 1) from frame A to frame B would yield:  $\approx (-0.354, 0.354, 0.866)$ .

A more detailed description including transformations between quaternions and rotation matrices (*direction cosines*) is given in Section 5.4 of [AD10]. See also [RD5].

## Appendix C

### Reference Frames

#### C.1 Definitions

The table below summarises the definitions of the relevant Swarm reference frames.

Name		Origin	Orientation	Description
ACC		ACC sensor	TBD	ACC sensor frame
ASM		ASM sensor	X along boom axis (forward) Y along S/C Y Z downwards (tilted)	ASM sensor frame
CRF		VFM sensor	Aligned with S/C axes, but fixed w.r.t. optical bench	Common reference frame of the STR. See Section C.2 below
GPSR		GPSR antenna reference center	TBD	Reference frame for the GPSR
ICRF		Barycenter of Solar System		Inertial, International Celestial reference frame
ITRF		Center of Earth	X along Greenwich meridian Y along 90° E meridian Z towards North pole	International Terrestrial reference frame
NEC		Reference position	N towards North E towards East C towards center of Earth	Local North-East-Center reference frame, position dependent. See Section C.3
S/C		Bottom center of face-plate	X nominal flight direction Y sideways (“right”) Z downwards (nadir)	S/C reference frame
S	CHU1	Center of corresponding CHU CCD	Z along boresight of CCD X and Y along CCD axes	Reference frames of the three STR CHUs
T	CHU2			
R	CHU3			
TIIH		Centers of TII H and V sensors	TBD	TII H-sensor frame
TIIV			TBD	TII V-sensor frame
VFM		VFM sensor	Z along boom axis (backwards) Y along S/C Y X downwards (tilted)	VFM sensor frame



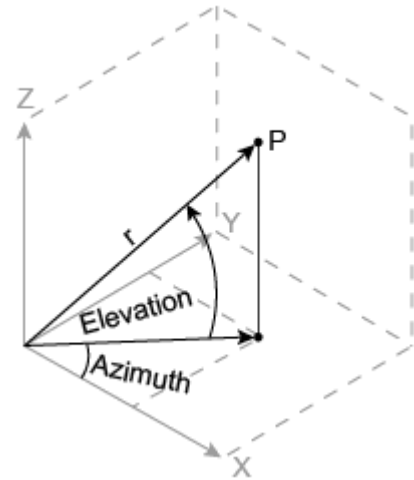
### Spherical coordinates

Spherical coordinates are given as radius ( $r$ ), elevation ( $\theta$ ), and azimuth ( $\varphi$ ) as depicted in the figure on the right. The quantities may be computed as:

$$r = \sqrt{x^2 + y^2 + z^2}$$

$$\theta = \text{atan2}(z, \sqrt{x^2 + y^2})$$

$$\varphi = \text{atan2}(y, x)$$



### C.2 STR Common Reference Frame (CRF)

The common reference frame of the STR is fixed w.r.t. the optical bench with origin at the center of the VFM sensor and oriented in the general direction of the S/C frame, i.e.  $\text{CRF}_x$  is in the general flight direction,  $\text{CRF}_z$  is in the general nadir direction, and  $\text{CRF}_y$  forms a right handed system.

### C.3 North East Center (NEC)

The NEC frame is defined in Section 3.7.2.3 of Swarm System Requirements Document, [SW-RS-ESA-SY-001]. At position  $\mathbf{r}$ , the NEC frame unit vectors are:

$$\mathbf{e}_{\text{center}} = -\mathbf{r} / |\mathbf{r}| \tag{C-10}$$

$$\mathbf{e}_{\text{east}} = \begin{cases} \mathbf{a} / |\mathbf{a}|, \mathbf{a} = \mathbf{e}_{\text{center}} \times (0\ 0\ 1)^T \\ (0\ 1\ 0)^T, \text{ if } |\mathbf{a}| = 0 \end{cases} \tag{C-20}$$

$$\mathbf{e}_{\text{north}} = \mathbf{e}_{\text{east}} \times \mathbf{e}_{\text{center}} \tag{C-30}$$

Hence, the rotation matrix,  $\mathbf{R}_{\text{NEC} \leftarrow \text{ITRF}}$ , is (with  $\mathbf{r}$  given in ITRF):

$$\mathbf{R}_{\text{NEC} \leftarrow \text{ITRF}} = \begin{pmatrix} \mathbf{e}_{\text{north}} \\ \mathbf{e}_{\text{east}} \\ \mathbf{e}_{\text{center}} \end{pmatrix} \tag{C-40}$$

## Appendix D

### B-Splines

A polynomial spline is a piecewise polynomial function. In its most general form a polynomial spline,  $S: [a,b] \rightarrow \mathfrak{R}$ , consists of polynomial pieces  $PP_i: [t_i, t_{i+1}] \rightarrow \mathfrak{R}$ , where  $a = t_0 < t_1 < t_2 < \dots < t_{k-1} < t_k = b$ . I.e.  $S(t) = PP_i(t)$  for  $t_i \leq t < t_{i+1}$ ,  $i = 0 \dots k-1$ . The given  $k+1$  points  $t_i$  are called knots.

A B-spline of degree  $n$  is composed of basis B-splines,  $b_{i,n}$ , of degree  $n$ :

$$S(t) = \sum_{i=0}^{k-n-1} P_i b_{i,n}(t), t \in [t_0, t_k] \quad (D-10)$$

The  $P_i$  are called *control points* or *de Boor points*. The  $m-n$  basis B-splines of degree  $n$  can be defined using the *Cox-de Boor recursion formula*

$$b_{i,0}(t) = \begin{cases} 1 & \text{if } t_i \leq t < t_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (D-20)$$

$$b_{i,j}(t) = \frac{t - t_i}{t_{i+j} - t_i} b_{i,j-1}(t) + \frac{t_{i+j+1} - t}{t_{i+j+1} - t_{i+1}} b_{i+1,j-1}(t) \quad (D-30)$$

Cubic B-splines with uniform knot-vector is the most commonly used form of B-spline. The *blending* function can easily be precalculated, and is equal for each segment in this case. Put in matrix-form, it is:

$$S_i(t) = \begin{pmatrix} t^3 & t^2 & t & 1 \end{pmatrix} \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ 3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{pmatrix} \begin{pmatrix} P_{i-1} \\ P_i \\ P_{i+1} \\ P_{i+2} \end{pmatrix} \quad (D-40)$$

Note though, this is not globally the case for the application here due to reduction of edge effects and potential gaps in the time series (see Section **Knots** below), hence the middle matrix of Eq. (D-40) must be computed from the preceding formulas directly.

#### 8.4.3.1 Derivative

The derivative of a degree  $n$  B-spline,  $dS(t)/dt$ , is a degree  $n-1$  B-spline. Let  $P'_i$  be the control points of  $dS(t)/dt$ , then:

$$P'_i = \frac{n(P_i - P_{i-1})}{t_{i+n} - t_i} \quad (D-50)$$

where  $P_{-1} = P_{k-n} \equiv 0$ .

#### 8.4.3.2 Knots

The knots for the cubic B-splines are constructed in a special way to avoid oscillation effects near the end points of the data segment. For smoothing B-splines, the knots consist of a number of repeated knots at each end and equidistantly spaced *interior* knots. For non-smoothing B-splines, i.e. when `CCDB...Spline_Pars.Knot_Space = 0`, the knots are identical

to the time-instants of the data samples with a reduced repetition at each end. In the event of gaps in the data segment, some knots may need to be removed; the algorithm for this is given at the end of this section.

Generally, let:

ks	Given knot spacing (= CCDB...Spline_Pars.Knot_Space)
$t_0, \dots, t_{n-1}$	Time-instants of data samples (including possible extensions)
T	Nominal sampling rate (_CCDB...Spline_Pars.T_nom)
r	Number of end-knot repetitions (see below)
$k_r, k_{r+1}, \dots, k_{m+r-2}, k_{m+r-1}$	Time-instants of knots. $k_0, \dots, k_{m-1}$ are the internal knots, i.e. $t_0 \leq k_0, k_i < k_{i+1}$ for $i = 0, 1, \dots, m-2$ , and $k_{m-1} \leq t_{n-1}$ ; the remaining knots are repeated end-knots, i.e. $k_i = t_0$ for $i < 0$ and $k_j = t_{n-1}$ for $j \geq m$ .

The B-spline knots for the non-smoothing B-spline, i.e. when  $ks = 0$ , are simply the time-instants of the extended data sequence. Hence,  $m = n$  and  $k_i = t_i$  for  $i = 0, 1, \dots, n-1$ . The number of end-knot repetition,  $r$ , is one. In summary, the knot sequence is:  $t_0, t_0, t_1, t_2, \dots, t_{n-1}, t_{n-1}$ .

For smoothing B-splines, the knot sequence consists of three-fold end-knots, i.e.  $r = 3$ , and interior knots are computed as follows:

$k_{rem}$	Remainder from given knot spacing: $k_{rem} = \text{mod}(t_{n-1} - t_0, ks)/2$
$k_{offset}$	Offset of first, interior knot: $k_{offset} = \begin{cases} k_{rem} + ks/2 & \text{if } k_{rem} + ks/2 \geq T \\ k_{rem} + ks & \text{otherwise} \end{cases}$
$t_i$	$= t_0 + k_{offset} + i T$ for $i = 0, 1, \dots, m_0-1$ , iff $n > 4$ (if $n \leq 4$ there are no interior knots)

The knot sequence has to be trimmed in case there are (small) gaps in the data sequence. First, all knots between  $t_0$  and  $t_1$  and between  $t_{n-2}$  and  $t_{n-1}$  are removed, that is, knots for which  $t_0 < k_i < t_1$  or  $t_{n-2} < k_i < t_{n-1}$  are removed leaving  $m \leq m_0$  interior knots. Next, some knots may need to be removed since there can be no more than 3 knots between data points and there can be no more than 3 more knots than data points to the left or right of any point on the time-line. This is ensured by the following algorithm where  $i$  is the index of  $t$  ( $i = 0 \dots n-1$ ) and  $j$  is the index of  $k$ ,  $j = -3, -2, \dots, m+2$  (note:  $t_{n-1} < k_{m+2}$ ):

```

lead = 0
i = 0
j = -3
j0 = j
while((i < n) and (j < m+3))
    if( $t_i \geq k_j$ )
        j = j + 1
        lead = lead + 1
    else

```

```

if(lead > 3)
    % There are lead-3 knots too many.
    Replace the knots  $k_{j_0}, \dots, k_{j-1}$  by  $p=(j-j_0-lead+3)$  new knots
    computed as the means of  $k_{j_0}, \dots, k_{j-1}$  divided in  $p$  parts
    (if  $p=3$  and  $\text{mod}(j-j_0,3)=1$  make middle section of  $k$ 's one
    larger, if  $\text{mod}(j-j_0,3)=2$  make outer sections one larger)
    lead = 2
elseif(lead > 0)
    % lead can not be negative
    lead = lead - 1
endif
j0 = j
i = i+1
endif
endwhile
if(lead + m + 3 - j > 3)
    Repeat the above backwards,
    i.e. start with  $i = n-1$  and  $j = m+2$  and count these downwards
    if(lead > 3) by the end of the while-loop
        replace the middle18 lead-2 interior knots by one knot =
        their
        mean
    endif
endif
endif

```

### 8.4.3.3 Collocation Matrix

The cubic B-splines used in this document are estimated by construction of collocation matrices via the `spco1` MATLAB function listed on the following pages.

### 8.4.3.4 Damping of Second Derivatives (Acceleration)

The matrix for the second derivative damping,  $\Lambda$ , is using equidistant time-instants ( $t_{acc}$ ) at  $1/5$  of the regular knot-spacing though at least the nominal sampling period, i.e.  $dt_{acc} = \max(ks/5, T)$ .  $t_{acc}$  is centered in the data interval  $t_0 \dots t_{n-1}$ , i.e.  $t_{acc_0} = t_0 + dt_{acc}$ , where  $dt_{acc} = \text{mod}(t_{n-1} - t_0, dt_{acc})/2$ , and  $t_{acc_i} = t_{acc_0} + i * dt_{acc}$ . If  $dt_{acc}$  is smaller than  $1/4$  of the nominal sampling rate,  $dt_{acc}$  is increased by  $dt_{acc}/2$ . Now the second differences of the rows of the collocation matrix, `spcol(knots, sp_order, t_acc)`, are divided by  $dt_{acc}^2$ , weighted, and squared to produce  $\Lambda$ . The weights are all one except near the edges where they decay slowly to zero over a period, denoted *width*, of 90 (nominal) samples. If the total period of  $t_0 \dots t_{n-1}$  is less than  $180 T + 2 dt_{acc}$ , *width* is reduced to  $(t_{n-1} - t_0)/2 - dt_{acc}$ . The weights,  $w_{ii}$ , are computed as follows with  $width_{max} = 90 * T$ .

$w_{ii}$	for ...
$(\frac{1}{2} - \cos(\pi * (t_{acc_i} - t_0)/width))/2 * width/width_{max}$	$t_{acc_i} \leq t_0 + width$
$(\frac{1}{2} - \cos(\pi * (t_{n-1} - t_{acc_i})/width))/2 * width/width_{max}$	$t_{acc_i} \geq t_{n-1} - width$
1	otherwise

<sup>18</sup> Towards lower indices, e.g. the “middle” 3 of 8 samples (numbered 1 through 8) would be samples 3, 4, and 5.

The complete calculation within MATLAB is as follows:

```

dt_acc = max(ks/5, T);
dt0_acc = mod(t(end) - t(1), dt_acc)/2;
if(dt0_acc < T/4)
    dt0_acc = dt0_acc + dt_acc/2;
end
t_acc = (t(1) + dt0_acc : dt_acc : t(end))';
if(numel(t_acc) > 2) % To be absolutely safe
    Gacc = diff(spcol(knots, sp_order, t_acc, 'sparse'), 2, 1) / dt_acc^2;
    width_max = 90 * T;
    width = min((t(end) - t(1))/2, width_max);
    t_acc([1 end]) = [];
    w_slow = (1 - cos(pi*(t_acc(t_acc <= t(1) + width) - t(1))/width))/2 *
        width / width_max;
    W = spdiags([w_slow; ...
        ones(size(Gacc,1) - 2*numel(w_slow), 1); ...
        w_slow(19min(size(Gacc,1)-end,end):-1:1)], ...
        0, size(Gacc,1), size(Gacc,1))
    Lambda = Gacc' * W * Gacc;
else
    Lambda = 0;
end

```

---

<sup>19</sup> ”min” function to safeguard against  $\text{size}(\text{Gacc},1) < 2 * \text{numel}(\text{w\_slow})$

```

function colloc = spcol(knots,k,tau,varargin)
%SPCOL B-spline collocation matrix.
%
% COLLOC = SPCOL(KNOTS,K,TAU) is the matrix
%
% [ D^m(i)B_j(TAU(i)) : i=1:length(TAU), j=1:length(KNOTS)-K ] ,
%
% with D^m(i)B_j the m(i)-fold derivative of B_j,
% B_j the j-th B-spline of order K for the knot sequence KNOTS,
% TAU a sequence of sites,
% both KNOTS and TAU are assumed to be nondecreasing, and
% m(i) is the integer #{ j<i : TAU(j) = TAU(i) }, i.e., the 'cumulative'
% multiplicity of TAU(i) in TAU.
%
% This means that the j-th column of COLLOC contains values and, perhaps,
% derivatives, at all the entries of the vector TAU, of the j-th
% B-spline of order K for the sequence KNOTS, i.e., the B-spline
% with knots KNOTS(j:j+K).
% The i-th row of COLLOC contains the value at TAU(i) of the m(i)-th
% derivative of all these B-splines, with m(i) the number of earlier
% entries of TAU that equal TAU(i) .
%
% Example:
% tau = [0,0,0,1,1,2]; % therefore, m equals [0,1,2,0,1,0]
% k = 3; knots = augknt(0:2,k); % therefore, knots equals [0,0,0,1,2,2,2]
% colloc = spcol(knots,k,tau)
%
% has the 6 entries of COLLOC(:,j) contain the value, first, and second
% derivative of B_j at 0, then the value and first derivative of B_j
% at 1, and, finally, the value of B_j at 2, with B_j the j-th B-spline of
% order k for the knot sequence knots; e.g., B_2 is the B-spline with
% knots 0,0,1,2.
%
% You can use COLLOC to construct splines with prescribed values and,
% perhaps, also some derivatives, at prescribed sites.
%
% Example:
% a = -pi; b = pi; tau = [a a a 0 b b]; k = 5;
% knots = augknt([a,0,b],k);
% sp = spmak(knots, ( spcol(knots,k,tau) \ ...
% [sin(a);cos(a);-sin(a);sin(0);sin(b);cos(b)] ) .' )
%
% provides the quartic spline, on the interval [a,b] with just one interior
% knot, at 0, that interpolates the sine function at a,0,b, but also matches
% its first and second derivative at a, and its first derivative at b .
%
% COLLOC = SPCOL(KNOTS,K,TAU,ARG1,ARG2,...) provides the same or a related
% matrix, depending on the optional arguments ARG1, ARG2, ... .
%
% If one of the optional arguments is 'slvblk', then COLLOC is in the al-
% most block-diagonal format (specialized for splines) required by SLVBLK.
%
% If one of the optional arguments is 'sparse', then COLLOC is a sparse
% matrix.
%
% If one of the optional arguments is 'noderiv', then multiplicities are
% ignored, i.e., m(i) = 0 for all i.
%
% The B-spline recurrence relations are used to generate the entries of the
% matrix.
%
% Example:
% t = [0,1,1,3,4,6,6,6]; x = linspace(t(1),t(end),101);
% c = spcol(t,3,x); plot(x,c)
%
% uses SPCOL to generate, in c(:,j), a fine sequence of values of the
% j-th quadratic B-spline for the given knot sequence t.

```

```

%
% See also SLVBLK, SPARSE, SPAPI, SPAP2, BSPLINE.

% Copyright 1987-2003 C. de Boor and The MathWorks, Inc.
% $Revision: 1.21 $

if ~isempty(find(diff(knots)<0))
    error('SPLINES:SPCOL:knotsdecreasing',...
        'The knot sequence KNOTS should be nondecreasing.')
end
if ~isempty(find(diff(tau)<0))
    error('SPLINES:SPCOL:TAUdecreasing', ...
        'The point sequence TAU should be nondecreasing.')
end

% Compute the number n of B-splines of order K supported by the given
% knot sequence and return an empty matrix in case there aren't any.

npk=length(knots); n=npk-k;
if n<1, warning('SPLINES:SPCOL:noBsplines', ...
    'There are no B-splines for the given input.')
    colloc = zeros(length(tau),0); return
end

% Settle the options:
slvblk=0; noderiv=0;
for j=4:nargin
    argj = varargin{j-3};
    if ~isempty(argj)
        if ischar(argj)
            if argj(1)=='s', slvblk=1;
                if length(argj)>1, if argj(2)=='p', slvblk=2; end, end
            elseif argj(1)=='n', noderiv=1;
            else error('SPLINES:SPCOL:wronginarg2',...
                ['The second optional argument should be ''s1'', ',...
                ''sp'', ''n'', or a number.'])
            end
        else
            switch j % for backward compatibility
            case 4, slvblk=1;
            case 5, noderiv=1;
            end
        end
    end
end

% If NODERIV==0, remove all multiplicities from TAU and generate repetitions
% of rows instead of rows containing values of successive derivatives.
nrows = length(tau); tau = reshape(tau,1,nrows);
if noderiv
    index = 1:nrows; m = ones(1,nrows); nd = 1; pts = tau;
else
    index = [1 find(diff(tau)>0)+1];
    m = diff([index nrows+1]); nd = max(m);
    if nd>k
        error('SPLINES:SPCOL:multtoohigh',...
            'Point multiplicity should not exceed the given order %g.',k);
    end
    pts = tau(index);
end

%set some abbreviations
kml = k-1;

% augment knot sequence to provide a K-fold knot at each end, in order to avoid
% struggles near ends of basic interval, [KNOTS(1) .. KNOTS(npk)] .
% The resulting additional B-splines, if any, will NOT appear in the output.

```

```

[augknot,add1] = augknt(knots,k); naug = length(augknot)-k;
pts = pts(:); augknot = augknot(:);

% For each i, determine savl(i) so that K <= savl(i) < naug+1, and,
% within that restriction,
%     augknot(savl(i)) <= pts(i) < augknot(savl(i)+1) .

savl = max(sorted(augknot(1:naug),pts), k);

b = zeros(nrows,k);

% first do those without derivatives
index1 = find(m==1);
if ~isempty(index1)
    ptls = pts(index1); savls = savl(index1); lpt1 = length(index1);
    % initialize the b array.
    lptls = index(index1); b(lptls,1) = ones(lpt1,1);

    % run the recurrence simultaneously for all ptl(i) .
    for j=1:kml
        saved = zeros(lpt1,1);
        for r=1:j
            tr = augknot(savls+r)-ptls;
            tl = ptls-augknot(savls+r-j);
            term = b(lptls,r)./(tr+tl);
            b(lptls,r) = saved+tr.*term;
            saved = tl.*term;
        end
        b(lptls,j+1) = saved;
    end
end

% then do those with derivatives, if any:
if nd>1
    indexm=find(m>1);ptss=pts(indexm);savls=savl(indexm);lpts=length(indexm);
    % initialize the bb array.
    %temp = [1 zeros(1,kml)]; bb = temp(ones(nd*lpts,1),:);
    bb = repmat([1 zeros(1,kml)],nd*lpts,1);
    lptss = nd*[1:lpts];

    % run the recurrence simultaneously for all pts(i) .
    % First, bring it up to the intended level:
    for j=1:k-nd
        saved = zeros(lpts,1);
        for r=1:j
            tr = augknot(savls+r)-ptss;
            tl = ptss-augknot(savls+r-j);
            term = bb(lptss,r)./(tr+tl);
            bb(lptss,r) = saved+tr.*term;
            saved = tl.*term;
        end
        bb(lptss,j+1) = saved;
    end

    % save the B-spline values in successive blocks in bb .

    for jj=1:nd-1
        j = k-nd+jj; saved = zeros(lpts,1); lptsn = lptss-1;
        for r=1:j
            tr = augknot(savls+r)-ptss;
            tl = ptss-augknot(savls+r-j);
            term = bb(lptss,r)./(tr+tl);
            bb(lptsn,r) = saved+tr.*term;
            saved = tl.*term;
        end
        bb(lptsn,j+1) = saved; lptss = lptsn;
    end

```



```

end

% now use the fact that derivative values can be obtained by differencing:

for jj=nd-1:-1:1
    j = k-jj;
    temp = repmat([jj:nd-1].',1,lpts)+repmat(lptsn,nd-jj,1); lptss=temp(:);
    for r=j:-1:1
        temp = repmat((augknot(savls+r)-augknot(savls+r-j)).'/j,nd-jj,1);
        bb(lptss,r) = -bb(lptss,r)./temp(:);
        bb(lptss,r+1) = bb(lptss,r+1) - bb(lptss,r);
    end
end

% finally, combine appropriately with b by interspersing the multiple
% point conditions appropriately:
dtau = diff([tau(1)-1 tau(:).' tau(nrows)+1]);
index=find(min(dtau(2:nrows+1),dtau(1:nrows))==0); % Determines all rows
                                                    % involving multiple tau.
dtau=diff(tau(index));index2=find(dtau>0)+1;      % We need to make sure to
index3=[1 (dtau==0)];                             % skip unwanted derivs:
if ~isempty(index2)
    index3(index2)=1+nd-m(indexm(1:length(indexm)-1));end
b(index,:)=bb(cumsum(index3),:);

% ... and appropriately enlarge savl
index = cumsum([1 (diff(tau)>0)]);
savl = savl(index);
end

% Finally, zero out all rows of b corresponding to TAU outside the basic
% interval, [knots(1) .. knots(npk)] .

index = find(tau<knots(1)|tau>knots(npk));
if ~isempty(index)
    b(index,:) = 0;
end

% The first B-spline of interest begins at KNOTS(1), i.e., at augknot(1+addl)
% (since augknot's first knot has exact multiplicity K). If addl<0,
% this refers to a nonexistent index and means that the first -addl columns
% of the collocation matrix should be trivial. This we manage by setting
savl = savl+max(0,-addl);

if slvblk % return the collocation matrix in almost block diagonal form.
    % For this, make the blocks out of the entries with the same
    % SAVL(i) , with LAST computed from the differences.
    % There are two issues, the change in the B-splines considered because of
    % the use of AUGKNOT instead of KNOTS , and the possible drop of B-splines
    % because the extreme TAU fail to involve the extreme knot intervals.

    % SAVL(j) is the index in AUGKNOT of the left knot for TAU(j) , hence the
    % corresponding row involves B-splines to index savl(j) wrto augknot, i.e.,
    % B-splines to index savl(j)-addl wrto KNOTS.
    % Those with negative index are removed by cutting out their columns (i.e.,
    % shifting appropriately the blocks in which they lie). Those with index
    % greater than n will be ignored because of last .

    last0 = max(0,savl(1)-max(0,addl)-k); % number of cols in trivial first block
    if addl>0 % if B-splines were added on the left, remove them now:
        width = kml+k;cc = zeros(nrows*width,1);
        index = min(k,savl-addl);
        temp = +repmat(nrows*[0:kml],nrows,1);
        cc(repmat([1-nrows:0]+nrows*index.',1,k)+repmat(nrows*[0:kml],nrows,1))=b;
        b(:)=cc(repmat([1-nrows:0].',1,k)+repmat(nrows*(k+[0:kml]),nrows,1));
        savl=savl+k-index;
    end
end

```

```

ds=(diff(sav1));
index=[0 find(ds>0) nrows];
rows=diff(index);
nb=length(index)-1;
last=ds(index(2:nb));
if addl<0 nb=nb+1; rows=[0 rows]; last=[last0 last]; end
if slvblk==1
  colloc=[41 nb rows k last n-sum(last) b(:).'];
else % return the equivalent sparse matrix (cf BKBRK)
  nr = (1:nrows).'; nc = 1:k; nrnc = nrows*k;
  ncc = zeros(1,nrows); ncc(1+cumsum(rows(1:(nb-1)))) = last;
  ncc(1) = last0; ncc = reshape(cumsum(ncc),nrows,1);
  ijs = [reshape(repmat(nr,1,k),nrnc,1), ...
         reshape(repmat(ncc,1,k)+repmat(nc,nrows,1), nrnc,1), ...
         reshape(b,nrnc,1)];
  index = find(ijs(:,2)>n);
  if ~isempty(index), ijs(index,:) = []; end
  colloc = sparse(ijs(:,1),ijs(:,2),ijs(:,3),nrows,n);
end
else % return the collocation matrix in standard matrix form
width = max([n,naug])+kml+kml;
cc = zeros(nrows*width,1);
cc(repmat([1-nrows:0].',1,k)+ ...
    repmat(nrows*savl.',1,k)+repmat(nrows*[-kml:0],nrows,1))=b;
% (This uses the fact that, for a column vector v and a matrix A ,
% v(A(i,j))=v(A(i,j)), all i,j.)
colloc = reshape(cc(repmat([1-nrows:0].',1,n) + ...
    repmat(nrows*(max(0,addl)+[1:n]),nrows,1)), nrows,n);
end

```

## Appendix E

### Miscellaneous Functions

This Appendix contains the description of some miscellaneous functions.

#### E.1 Table Look-Up Function, $\text{Interpol}_{\text{Lin}}$

The linear interpolation function,  $\text{Interpol}_{\text{Lin}}$ , is defined in the following for one and two dimensions. It uses the LUT1d and LUT2d look-up tables defined in [AD03]:

<i>Id</i>	<i>Name</i>	<i>Type</i>	<i>Description</i>
LUT1d	<i>One-dimensional look-up table</i>		
	m	integer	Number of entries in x and v.
	x	vector, length m	Nodes (keys)
	v	vector, length m	Function values
LUT2d	<i>Two-dimensional look-up table</i>		
	n	integer	Number of entries in y and lut
	y	vector, length n	Nodes (keys)
	Lut	vector, length n	1-d look-up tables

Now:

$$\text{Interpol}_{\text{Lin}}(\text{T1}, x') = v_i + \frac{x' - x_i}{x_{i+1} - x_i} (v_{i+1} - v_i) \quad (\text{E-10})$$

where

- T1 is of type LUT1d
- $x'$  is the key-value for the look-up
- x and v are elements (vectors) of T1
- i is the index such that  $x_i \leq x' \leq x_{i+1}$ , it is an error if i does not exist

The two dimensional case is very similar:

$$\text{Interpol}_{\text{Lin}}(\text{T2}, x', y') = F_i + \frac{y' - y_i}{y_{i+1} - y_i} (F_{i+1} - F_i) \quad (\text{E-20})$$

where

- T2 is of type LUT2d
- $x'$  and  $y'$  are the key-values for the look-up
- y are the nodes (keys) of T2
- i is the index such that  $y_i \leq y' \leq y_{i+1}$ , it is an error if i does not exist
- $F_i = \text{Interpol}_{\text{Lin}}(\text{T2.Lut}_i, x')$  is the values of one-dimensional look-ups

## E.2 Symmetric Finite Impulse Response Filter, Filter

Filter is the function applying a given set of filter coefficients (FIR\_FP) onto a time-series. A range check and an outlier or jump detection method is build into Filter. Generally, the time-series shall be equidistantly spaced in order to preserve the filter characteristics; hence gaps in the time-series – up to a specified length – are filled by linear interpolation. The filter coefficients are defined in [AD03]:

<i>Id</i>	<i>Name</i>	<i>Type</i>	<i>Description</i>
FIR_FP	<i>Finite Impulse Response filter coefficients, symmetric</i>		
	<b>n</b>	Integer, $\geq 0$	Number of entries in <b>b</b> ; $n = 0$ means no filter
	<b>b</b>	vector, length $n$	Filter coefficients, right-hand side, $b_2, \dots, b_n$ are mirrored to left-hand side coefficients
	<b>lim</b>	real	Threshold for outlier or jump detection
	<b>range</b>	vector	Allowed range of input data
	<b>T_nom</b>	real	Nominal sampling period, seconds
	<b>Max_Gap_Size</b>	real	Maximum length of data gap to be filled by interpolation, seconds

Now,  $\text{Filter}(\text{FIR\_FP}, \mathbf{t}^{\text{in}}, \mathbf{x}^{\text{in}})$  returns the time-series  $y$ :

$$y_i = \frac{b_1 x'_i + \sum_{j=2}^{n'} b_j (x'_{i+j-1} + x'_{i-j+1})}{b_1 + 2 \sum_{j=2}^{n'} b_j} \quad (\text{E-100})$$

where

$\mathbf{t}^{\text{in}}$  and  $\mathbf{x}^{\text{in}}$  are the input vectors of data time-instants and values

$i = 1, 2, \dots, k$ ;  $k$  is the length of the equidistant, gap-filled time-series  $\mathbf{t}, \mathbf{x}$  (and hence also the length of  $y$ )

**b** is the filter coefficients of FIR\_RP

$\mathbf{t}$  is the vector of output time instants, i.e. a version of  $\mathbf{t}^{\text{in}}$  with gaps shorter than **Max\_Gap\_Size** filled with equidistant points. Formally, let  $\tilde{\mathbf{t}} \subseteq \mathbf{t}^{\text{in}}$  be the time-instants of  $\tilde{\mathbf{x}}$  (see below), then  $\tilde{\mathbf{t}} \subseteq \mathbf{t}$ , and for  $i \geq j, i = 1 \dots k$ :

$$t_i = \begin{cases} \tilde{t}_j + \frac{m}{m_0} (\tilde{t}_{j+1} - \tilde{t}_j) & \text{if } \tilde{t}_{j+1} - \tilde{t}_j > 1.5 T_{\text{nom}} \text{ and } \tilde{t}_{j+1} - \tilde{t}_j \leq \text{Max\_Gap\_Size,} \\ \tilde{t}_j & \text{otherwise} \end{cases} \quad \text{where } m_0 = \left\lfloor \frac{\tilde{t}_{j+1} - \tilde{t}_j}{T_{\text{nom}}} + 0.5 \right\rfloor, m = 0 \dots m_0 - 1 \quad (\text{E-105})$$

$\mathbf{x}'$  is the vector of gap-filled data corresponding to the time-instants  $\mathbf{t}$ .

$\mathbf{x}'$  is computed by linear interpolation in time (to the instants  $\mathbf{t}$ ) of  $\mathbf{x}$ , where  $\mathbf{x}$  are the “good” values of  $\mathbf{x}^{\text{in}}$ . Let  $\tilde{\mathbf{x}} \subseteq \mathbf{x}^{\text{in}}$  for which  $\tilde{x}_i \in \mathbf{range}$  then  $\mathbf{x}$  is the subset of

$\tilde{x}$  for which (the second part of (E-110) distinguishes between single outliers and jumps):

$$|\tilde{x}_i - \tilde{x}_{i-1}| \leq \text{lim} \text{ or } |\tilde{x}_{i+1} - \tilde{x}_{i-1}| > \text{lim} \quad (\text{E-110})$$

$n'$  is the *running* filter width, which is reduced compared to  $n$  (of FIR\_FP) near the ends of the time-series and around jumps within the time-series. If there are no jumps:

$$n' = \begin{cases} i & \text{if } i < n \\ n & \text{if } n \leq i \leq k - n + 1 \\ k - i + 1 & \text{if } i > k - n + 1 \end{cases} \quad (\text{E-120})$$

If there is a jump from  $\tilde{x}_{m-1}$  to  $\tilde{x}_m$ , i.e. if  $|\tilde{x}_m - \tilde{x}_{m-1}| > \text{lim}$  and  $|\tilde{x}_{m+1} - \tilde{x}_{m-1}| > \text{lim}$ , then  $n'$  is defined near  $m$  as:

$$n' = \begin{cases} m - i & \text{if } m - n < i < m \text{ and } i \geq m/2 \\ i - m + 1 & \text{if } m \leq i < m + n - 1 \text{ and } i \leq (k + m)/2 \end{cases} \quad (\text{E-130})$$

with  $m$  being the index of the jump closest to  $i$ . The two last conditions of Eq. (E-130) ensure consistency near the ends of the time-series – if nearer the ends than any jump, Eq. (E-120) is used.

If  $n = 0$ ,  $y_i = x'_i$  (also the case, if  $n = 1$ ).

If any gap or outlier is detected, all affected  $y_i$  are flagged, i.e. all  $y_i$  which are  $n$  samples or closer to the gap/outlier. If a jump is detected, all affected  $y_i$  are flagged (with a different flag).

Filter(FIR\_FP,  $x$ ) may also be applied to time-series of vectors in which case Eq. (E-100) is applied to each component of  $x$  individually.

### E.3 S/C Mass and CoG Calculation

The functions to calculate the S/C mass and CoG are given below.

#### Input Data

The following input data from Level 1a.AOCS.Prop\_HK and Level 1a.AOCS.Thru\_HK are used:

- $P_A$  – Pressure of tank A
- $P_B$  – Pressure of tank B
- $V_{A,j}$  – Thermistor voltages, tank A,  $j = 1, \dots, 5$
- $V_{B,j}$  – Thermistor voltages, tank B,  $j = 1, \dots, 5$
- $V_i$  – Reference voltage
- $t_{thrus}$  – Time of last thruster operation

The following characterization parameters are used:

Element	Type	Description
CCDB. Bus. Tank	Thermistor	Array of calibration coefficients for the tank thermistors, order: A1, B1, A2, B2, ..., A5, B5
	K1	real Thermistor circuit calibration factor. Volt
	K2	real Thermistor circuit calibration factor. Ohm/Volt
	K3	real Thermistor circuit calibration factor. Volt
	a	real Steinhart coefficient 1
	b	real Steinhart coefficient 2
	c	real Steinhart coefficient 3
Vcal	real Nominal reference voltage	
CCDB. Structure. SC_Mass	Dry	real Spacecraft dry Mass
	Fuel	LUT2d Look-up table providing mass of fuel depending on pressure and temperature.  x: pressure [bar] y: temperature [°C] v: mass [kg]
	wT	vector Weights for temperature average, length 5
	a	real Filter coefficient for mass
CCDB. Structure. SC_CoG	Dry	vector CoG of dry spacecraft
	TankA	vector CoG of dry tank A
	TankB	vector CoG of dry tank B
	dL	real Variation of tank CoG in x-direction depending on pressure

The Level 1a.AOCS.Xxx\_HK, CCDB.Structure.SC\_Yyy, and CCDB.Bus.Tank.Thermistors prefixes are omitted in the following if the element is unambiguous.

### Algorithm

Compute the tank temperatures ( $X = A$  and  $B$ ,  $j = 1, 2, \dots, 5$ ):

$$T_{j,X} = 1 / (a + b \ln(R_{j,X}) + c (\ln(R_{j,X}))^3) - 273.15 \quad (\text{E-194})$$

where

$$R_{j,X} = K2 \, dV_{j,X} / (K1 - dV_{j,X}) \quad (\text{E-196})$$

and

$$dV_{j,X} = V_{X,j} - (V_i - \text{CCDB.Bus.Tank.Vcal}) - K3 \quad (\text{E-198})$$

Calculate weighted tank temperatures ( $X=A$  and  $B$ ):

$$T_X = \frac{\sum_{j=1}^5 wT_j T_{j,X}}{\sum_{j=1}^5 wT_j} \quad (\text{E-200})$$

Calculate fuel mass per tank by linear interpolation in look-up table ( $X=A$  and  $B$ ):

$$\tilde{m}_X = \text{Interpol}_{\text{Lin}}(\text{Fuel}, P_X, T_X) \quad (\text{E-210})$$

Calculate fuel mass

$$\tilde{m}_{fuel} = \tilde{m}_A + \tilde{m}_B \quad (\text{E-220})$$

The mass should not have changed since time of last thruster operation (i.e. variations are mainly due to HK data noise). In order to reduce this noise filter fuel mass using parameter  $a$  from CCDB.SC\_Mass as follows

$$m_{fuel} = \begin{cases} \tilde{m}_{fuel} & \text{if first packet after thruster operation} \\ a m_{fuel,previous} + (1-a) \tilde{m}_{fuel} & \text{otherwise} \end{cases} \quad (\text{E-230})$$

Adapt tank  $X$  ( $A$  and  $B$ ) masses accordingly

$$m_X = \frac{m_{fuel}}{\tilde{m}_{fuel}} \tilde{m}_X \quad (\text{E-240})$$

Calculate spacecraft mass:

$$\text{Mass} = m_A + m_B + \text{CCDB.SC_Mass.Dry} \quad (\text{E-250})$$

Calculate centre of gravity of tanks (X=A and B):

$$\mathbf{CoG}_X = \text{CCDB.SC\_CoG.TankX} - (\text{CCDB.SC\_CoG.dL P}_X \ 0 \ 0)^T \quad (\text{E-260})$$

Calculate centre of gravity of spacecraft:

$$\mathbf{r}_{\text{CoG}} = \frac{\text{SC\_Mass.Dry SC\_CoG.Dry} + m_A \mathbf{CoG}_A + m_B \mathbf{CoG}_B}{\text{Mass}} \quad (\text{E-270})$$



## E.4 Aberrational Correction

The method for the correction of the relativistic aberration is described in “Attitude Correction for Relativistic Aberration” [Attitude Correction for Relativistic Aberration.pdf](#) (next pages). The inertial reference frame ( $v s n$ ) – Vernal, Summer, North – is orientated identical to ( $x y z$ ) of the ICRF. The velocity vector of the observer is with respect to the Stars, i.e. it shall include S/C orbit velocity as well Earth velocity.

## Attitude Correction for Relativistic Aberration

Troelz Denver, August 2005

### Background

When a scene (e.g. a star field) is observed by an observer moving at high velocities tangential to the scene direction, the apparent position of the scene objects (e.g. stars) will shift towards the motion direction. The measurements of a sensor determining the position of scene objects will therefore be shifted in the forward direction of motion. On the contrary, the measurements of a sensor determining its own position based on scene objects will be shifted in the backward direction. When the sensor velocity is not orthogonal to the observed scene, the correction magnitude will decrease by sinus to the angle between the velocity vector and the vector to the scene.

### Procedure

Let an attitude of a sensor,  $A$ , be defined as a orthogonal 3 x 3 matrix:

$$\bar{A} = [\bar{x} \quad \bar{y} \quad \bar{z}] = \begin{bmatrix} vx & vy & vz \\ sx & sy & sz \\ nx & ny & nz \end{bmatrix}$$

such that the three columns in  $A$  defines the orientation of the sensor axes (called  $x$ ,  $y$  and  $z$ ) in an inertial reference frame with axes called  $v$ ,  $s$  and  $n$ . The sensor frame is defined such that the  $z$ -axis is equal to the pointing direction and the  $x$  and  $y$  are selected such that an arbitrary right-handed coordinate system is realized.

Furthermore, let the velocity of the sensor,  $v$ , be defined as:

$$\vec{v} = \begin{bmatrix} vv \\ vs \\ vn \end{bmatrix}$$

such the three elements define the velocities along each of the three axes of the inertial reference frame. The element units are m/s.

First of all, a vector to perform the correction about,  $v_r$ , is derived as the cross product between the observation direction and the velocity vector.

$$\vec{v}_r = \frac{\vec{v} \times \bar{z}}{c}$$

where the speed of light,  $c$ , is given in the same units as the elements of  $v$  (m/s). Since  $|v_r|$  is proportional to sinus to the angle between  $v$  and  $z$ , the magnitude dependency of this angle is already included in the figure.

In the following description, it is applied that the rotations about different system axes do commute for small angular motions. The correction for each of the system axes is performed. The new  $z$ -axis is found from adding the correction to the original axis:

$$\vec{z} = \vec{z} + \vec{z} \times \vec{v}_r$$

Similar, the two other system axes will be corrected accordingly:

$$\vec{x} = \vec{x} + \vec{x} \times \vec{v}_r$$

and

$$\vec{y} = \vec{y} + \vec{y} \times \vec{v}_r$$

The attitude at this stage formed by the three independently corrected system axes will not be orthogonal. Again, it is used that rotations about system axes will commute for small angular motions.

First of all, the corrected attitude  $z$ -axis is formed by the estimated  $z$ -axis by a simple normalization:

$$\vec{z}' = \frac{\vec{z}}{|\vec{z}|}$$

The corrected attitude  $y$ -axis is formed by the  $z$ -axis and the previously estimated  $x$ -axis:

$$\vec{y}' = \frac{\vec{z}' \times \vec{x}}{|\vec{z}' \times \vec{x}|}$$

Finally, the corrected attitude  $x$ -axis is found to realize the right-handed coordinate frame:

$$\vec{x}' = \vec{y}' \times \vec{z}'$$

The corrected attitude,  $A'$ , given as

$$\vec{A}' = [\vec{x}' \quad \vec{y}' \quad \vec{z}']$$

will now be orthogonal.

## Appendix F

### ASM Algorithm Descriptions

The description of the Swarm ASM Level 1b processing algorithms supplied by the instrument manufacturer is contained in [AD05]: “Swarm ASM level 1B algorithms”, [SW-CEA-AUT-ALGO1B-001 4-0.pdf](#).

## Appendix G

### VFM Algorithm Description

The description of the Swarm VFM instrument and the Level 1b processing algorithms supplied by the instrument manufacturer is contained in [AD06]: “Vector Field Magnetometer for the Swarm Project, Users Manual”, [SW-MA-DTU-VF-5001\\_2\\_UserManual.pdf](#)

## Appendix H

### STR Algorithm Description

The description of the Swarm STR instrument and the Level 1b processing algorithms supplied by the instrument manufacturer is contained in [AD07]: “micro Advanced Stellar Compass User’s Manual”, [ASC-DTU-MA-3001 v2.0 User Manual.pdf](#).

## Appendix I

### EFI-TII Algorithm Description

The Swarm EFI instrument package consists of the Thermal Ion Imager (TII) and the Langmuir Probe (LP) instruments. The description of the TII Level 1b processing algorithms supplied by the instrument manufacturer is contained in [AD08]: “Swarm CEFI Level 1b Processor Definition“, [SW-TN-UOC-EF-003.pdf](#), version 1.14.

## Appendix J

### EFI-LP Algorithm Description

The Swarm EFI instrument package consists of the Thermal Ion Imager (TII) and the Langmuir Probe (LP) instruments. The description of the LP Level 1b processing algorithms supplied by the instrument manufacturer is contained in [AD09]: “Swarm CEFI-LP Level 1b Algorithms”, [SW-TN-IRF-EF-003\\_level1b\\_algorithms.pdf](#), version 2.6.



## Appendix K

### GPSR Algorithm Description

The description of the Swarm GPSR Level 1b processing algorithms supplied by the instrument manufacturer is contained in [AD11]: “Swarm GPSR TE-12 Instrument L1b Algorithms Definition”, [SW-TN-SES-GP-0018\\_6.pdf](#).

## Appendix L

### ACC Algorithm Description

The description of the Swarm ACC instrument Level 1b processing algorithms supplied by the instrument manufacturer is contained in [AD13]: “Accelerometer Instrument Level 1b Algorithms Definition”, [SW-DF-VZL-AC-0001\\_3\\_4\\_L1B\\_Algorithm.pdf](#).